the net beyond the web

libreplanet 2022

bandali@gnu.org

goal: share some thoughts, spark discussion

net ?= web

[inter]net: interconnected network of networks

where

the networks and the devices comprising them
use tcp/ip to communicate with each other

web: is part of the internet

consists of websites, like

    https://www.gnu.org

which consist of webpages, like

    https://www.gnu.org/gnu/about-gnu.html

which are written in html

```
html + css



... + javascript
```

web initially a method/protocol for sharing

documents, information, media, ...

example: the first website (still accessible):

http://info.cern.ch/hypertext/WWW/TheProject.html

aside: shout out and thanks to

wayback machine - https://web.archive.org

early on:

small websites consisting entirely of text

and a bit later images/photos as well

joan stark (jgs)

https://en.wikipedia.org/wiki/Joan_Stark

Veronica Karlsson (vk)

https://web.archive.org/web/20160319004113/http://www.ludd.luth.se/~vk/

javascript

[before] javascript:

webpages were fully static

this was not desired by web developers

netscape decided to add a scripting language
to their web browser, netscape navigator

netscape hired Brendan Eich to embed the
scheme language into their browser

yay!!

right?

... right??

nestscape collaborated with sun microsystems
to embed java into their browser

netscape management decided the new scripting
language must have a syntax similar to java
rather than scheme

Eich designed and implemented javascript in 10 days

no shortage of interesting takes and/or criticisms
of javascript as a language, and its impacts

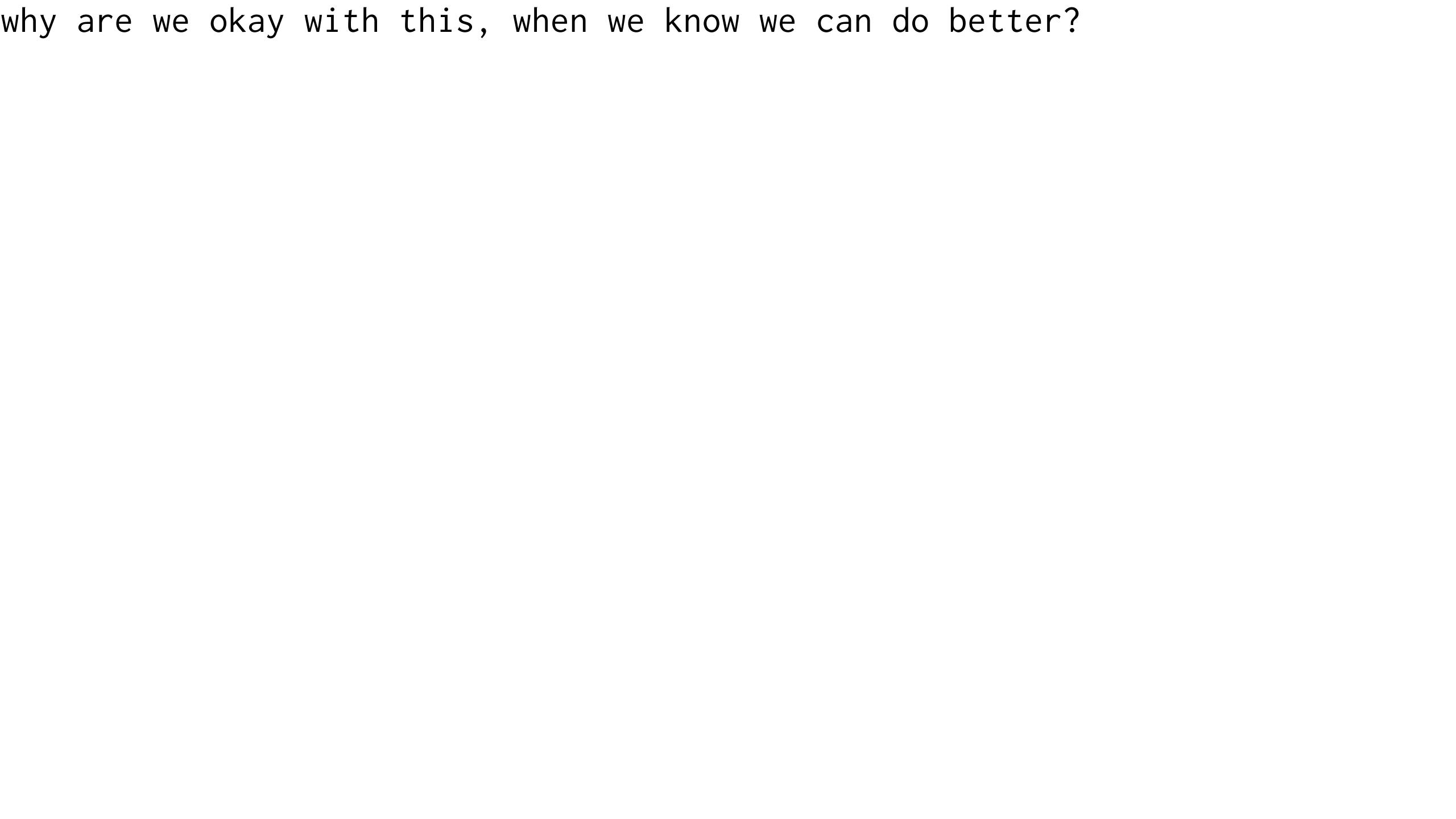"the birth & death of javascript" and "wat"
talks by Gary Bernhardt

other interesting reads, such as

https://wiki.haskell.org/The_JavaScript_Problem
https://en.wikipedia.org/wiki/JavaScript#Security
https://en.wikipedia.org/wiki/Browser_security
https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=javascript

text --> multimedia --> ... programs that run on your machine

... and they are virtually everywhere!

the wwworst app store
by Alexandre Oliva

https://www.gnu.org/philosophy/wwworst-app-store.html

why are we okay with this, when we know we can do better?

how gnu/linux (and other unix-like) distributions
handle distributing of software to their users

we need similar control/trust/consent in the browser

and a shift in the norms and status quo among web developers

for instance by creating and documenting publicly-accessible
api endpoints, so that anyone could write and use a custom
application for communicating with their website/service

"taking back the web with haketilo"
libreplanet 2022 (day 1) talk
by Nicholas Johnson and Wojciech Kosior

https://libreplanet.org/2022/speakers/#5790
https://hydrillabugs.koszko.org/projects/haketilo

what about other protocols?

gopher, gemini, spartan, ...

http(s)

one idea:

plain text + some rules + convenience (via extension)

some *bold* text, _emphasized_ text,

a dashed title

[image-filename.png]
 and a simple caption

a link to the <contact.txt> page

> a tiny block of
> quoted text

    (message "and a 4-space indented code block")
    (call-interactively #'butterfly) ; M-x butterfly

```
(eqn "sqrt(b^2-4ac)")

(eqn
 (sqrt (- (expt b 2) (* 4 a c))))

\sqrt{b^2-4ac}
```

do we want a programming language?

maybe, maybe not. if we do,

1. be used only when really needed

2. code blocks be clearly & freely licensed

3. any code executed only with user's consent,
   potentially along with a permission system
   that gives the user fine control over what
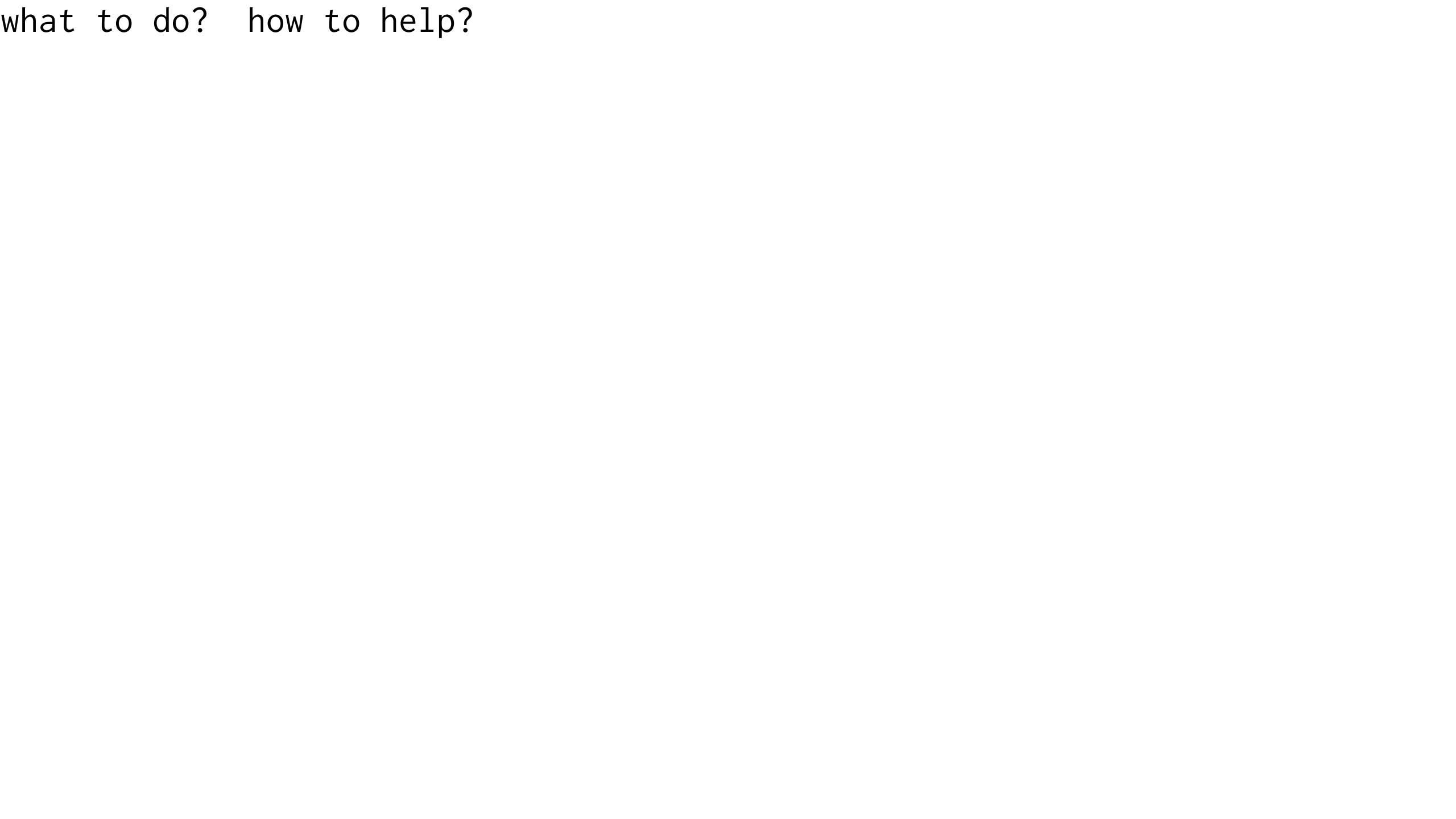   functions can execute for each site

user should have ultimate control
over how documents look

http does not have to be the only transport

people have been thinking about or experimenting
with other transport methods/protocols

https://alexschroeder.ch/wiki/2022-02-08_NNCP_distributed_text

gopher://zaibatsu.circumlunar.space/0/~solderpunk/phlog/low-budget-p2p-conte
nt-distribution-with-git.txt
https://gopher.tildeverse.org/zaibatsu.circumlunar.space/0/~solderpunk/phlog
/low-budget-p2p-content-distribution-with-git.txt

what to do? how to help?

as a user:

- use freedom- and privacy-enhancing
  browser extensions such as:
  gnu.org/s/librejs, jshelter.org,
  haketilo, and ublock origin

  (see "taking back the web with haketilo"
   by Nicholas and Wojciech for a list of
   even more such extensions)

- say no to nonfree javascript
  "saying no even once helps"
  https://www.gnu.org/philosophy/saying-no-even-once.html

- check out and participate in
  small web or small net initiatives,
  as well as tilde or pubnix communities
  like tilde.team and sdf.org

as a developer:

- release and label your javascript as free software
  https://www.gnu.org/software/librejs/free-your-javascript.html

- check out protocols like gopher and gemini,
  see what they have to offer, and think about
  what an http-based approach could look like

- contribute:
  - other browsers like luakit, netsurf, and visurf
  - help maintain liberated and privacy-enhanced
    versions of firefox like gnu icecat and abrowser

- don't use javascript unless you really need to.
  when you do, please license it freely, and follow
  principles like 'progressive enhancement' to make
  sure your website works without javascript and css

if you happen to work on either side of the
current web browser duopoly, or work on web
standards, please consider:

do we really *need* all the new complexity?

12 years ago, mozilla removed support for
gopher on the grounds of reducing complexity
and attack vectors. do the current web and
its browsers resemble anything but complexity?

firefox and chromium are now each more than
30 million lines of code, on the same order
of magnitude and in fact currently larger than
the kernel linux. new code and complexity
keeps being added, while at the same time,
user freedom, control, and often privacy seem
to be gradually lessened and eroded.

the net i'd like to see beyond the current web

1. does not facilitate tracking users;

2. has freedom, control, and consent built in; and

3. is in many ways simpler, has a reasonable and
   finite scope, and has several server and client
   implementations.

thanks!

bandali@gnu.org
kelar.org/~bandali

references
----

http://info.cern.ch/hypertext/WWW/TheProject.html
https://web.archive.org
https://en.wikipedia.org/wiki/Joan_Stark
https://web.archive.org/web/20160319004113/http://www.ludd.luth.se/~vk/
https://wiki.haskell.org/The_JavaScript_Problem
https://en.wikipedia.org/wiki/JavaScript#Security
https://en.wikipedia.org/wiki/Browser_security
https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=javascript
https://www.gnu.org/philosophy/wwworst-app-store.html
https://libreplanet.org/2022/speakers/#5790
https://hydrillabugs.koszko.org/projects/haketilo
https://www.gnu.org/philosophy/javascript-trap.html
https://www.gnu.org/philosophy/who-does-that-server-really-serve.html
https://www.fsf.org/campaigns/freejs
https://en.wikipedia.org/wiki/Gopher_(protocol)
https://gopher.floodgap.com
https://gemini.circumlunar.space
https://spartan.mozz.us
http://www.aaronsw.com/weblog/000574
http://www.aaronsw.com/2002/rss30
https://alexschroeder.ch/wiki/2018-12-01_Thinking_about_the_real_RSS_3.0
https://alexschroeder.ch/wiki/2022-02-08_NNCP_distributed_text
gopher://zaibatsu.circumlunar.space/0/~solderpunk/phlog/low-budget-p2p-content-distribution-with-git.txt
https://gopher.tildeverse.org/zaibatsu.circumlunar.space/0/~solderpunk/phlog/low-budget-p2p-content-distribution-with-git.txt
https://www.gnu.org/philosophy/saying-no-even-once.html
https://tilde.team
https://sdf.org
https://www.gnu.org/software/librejs/free-your-javascript.html
https://www.gnu.org/distros/free-system-distribution-guidelines.html
https://bugzilla.mozilla.org/show_bug.cgi?id=388195
https://drewdevault.com/2020/03/18/Reckless-limitless-scope.html
https://drewdevault.com/2021/09/11/visurf-announcement.html
https://drewdevault.com/2021/09/27/Let-distros-do-their-job.html
https://en.wikipedia.org/wiki/Progressive_enhancement

https://kelar.org/~bandali/2022/03/20/net-beyond-web.html
gopher://kelar.org/0/~bandali/2022/03/20/net-beyond-web.txt