

the real story behind Wayland and X

or:

every single LWN comment is wrong



(except mine.)



'let me summarize every
wayland discussion on the internet:
I'VE SEEN A WINDOW SYSTEM
SO I KNOW HOW THEY SHOULD WORK
PAY ATTENTION TO MEEEEEEE'

— Adam Jackson



wait, who are you?



older than X11

younger than X10



started packaging XFree86 in 2002

joined X.Org at my first LCA in 2004

built the first modular X.Org server



Australia was too hot, moved to Helsinki
(in February, having seen snow once)



worked at Nokia on:

770, N800, N810, N900, N9



this was the state of the art in 2006



NOKIA

Home

Updated: 16 May

Cricket: Aussie
Brazil targets 2
Arcelor strengt
Major melt' for
Bahrain suspe
India airports agree...
Hope over self-focusing
Antarctic birds 'breedi...
Kong DVD breaks studi...
Alcatel merger 'will sav...
America Online to rebr...

Connected

✓ On
Away
Invisible
Off

Presence settings...

uradio

Anna
Bertha
Patricia
David.K

Tuesday
16 May 2006

no acceleration

gtk+ 2.6

three-level submenus



... and RealPlayer/Helix



wrote input hotplug support

D-Bus/HAL support (sorry ...)



(some months pass ...)





surprise!



oh dear.



overnight career change:

much less keyboards

much more fixing visual glitches



for the next five years.





PASSPORT

Authority / Autorité
2014
AUSTRALIA



PASSPORT

Authority / Autorité
2009
2019
AUSTRALIA



PASSPORT

WITH NECTAR YOU WO
POINTS

but now I work on Wayland!



hey internet peanut gallery



krh was an X developer too



but enough about me



what is X11?



classic XII I



mechanism, not policy



a mechanism for any client to
enforce its idiot wishes on everyone



immediate-mode rendering

draw rectangles!

draw text!

blit images!



basic window management

reparent windows!

move windows!

window borders — solid OR dashed!



one keyboard

one three-button mouse



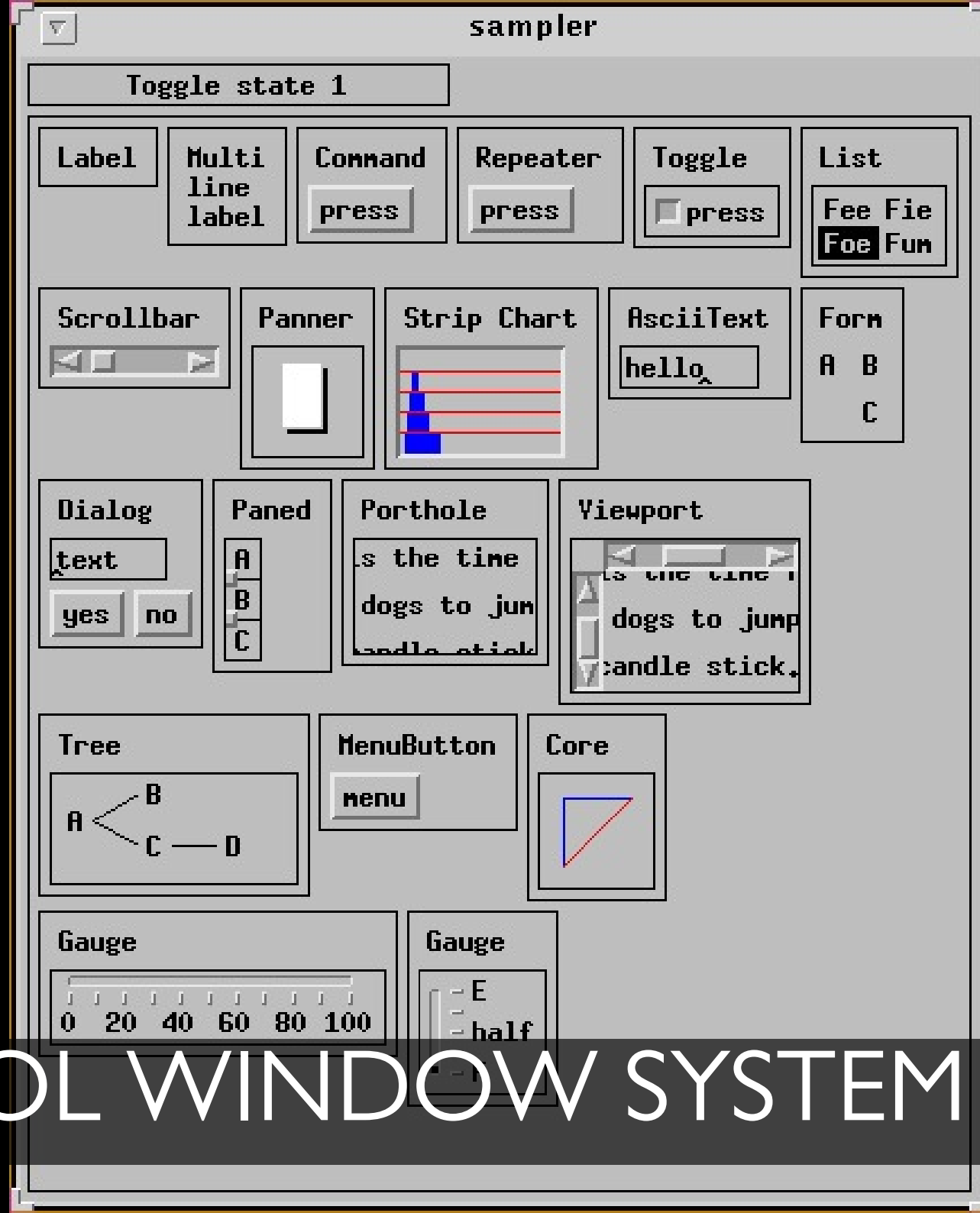
network transparent!

(hi LWN)



pretty straightforward really





COOL WINDOW SYSTEM BRO

[time passes]



XFree86



hardware got complicated



multiple input devices — kinda
keyboards with bells on — literally
multiple GPUs — again



rendering got complicated too



OpenGL

accelerated video

tasteful & minimal themes



window management got awful



multiple desktops

weird and wonderful window types

shaped windows?!



everything got out of hand



'Programming X is like reading one of those French philosophers where afterwards you start wondering whether you really know anything for sure.'

— Thomas Thurman



at least 25 more extensions
thousands more pages of spec



thanks to politics,
never touched the core protocol
or core server code



working around deficiencies



not fixing them.



they also worked around poor OSes



ran video BIOSes

... sometimes through an x86 emulator

performed system power management



legacy I/O port management

PCI bus management — in assembly!

binary loader — ELF, COFF, a.out



the X server became its own OS.



the dumbest OS you've ever seen.



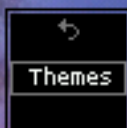
could generate a config file for you
not smart enough to just *(!@# use it



life was, basically, terrible



WED 30 MAR
10:57 PM
10:37
23:15
10.9M/s
+20 C 68%
98%
CPU
108 PROCS
2 USERS
Proc
229
ppp0
11:33
246M -
375M -
68 22K
101.30
Vol
Pcm
http://
CB
xmms

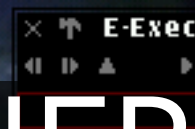
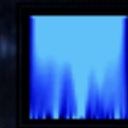
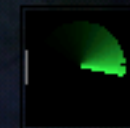
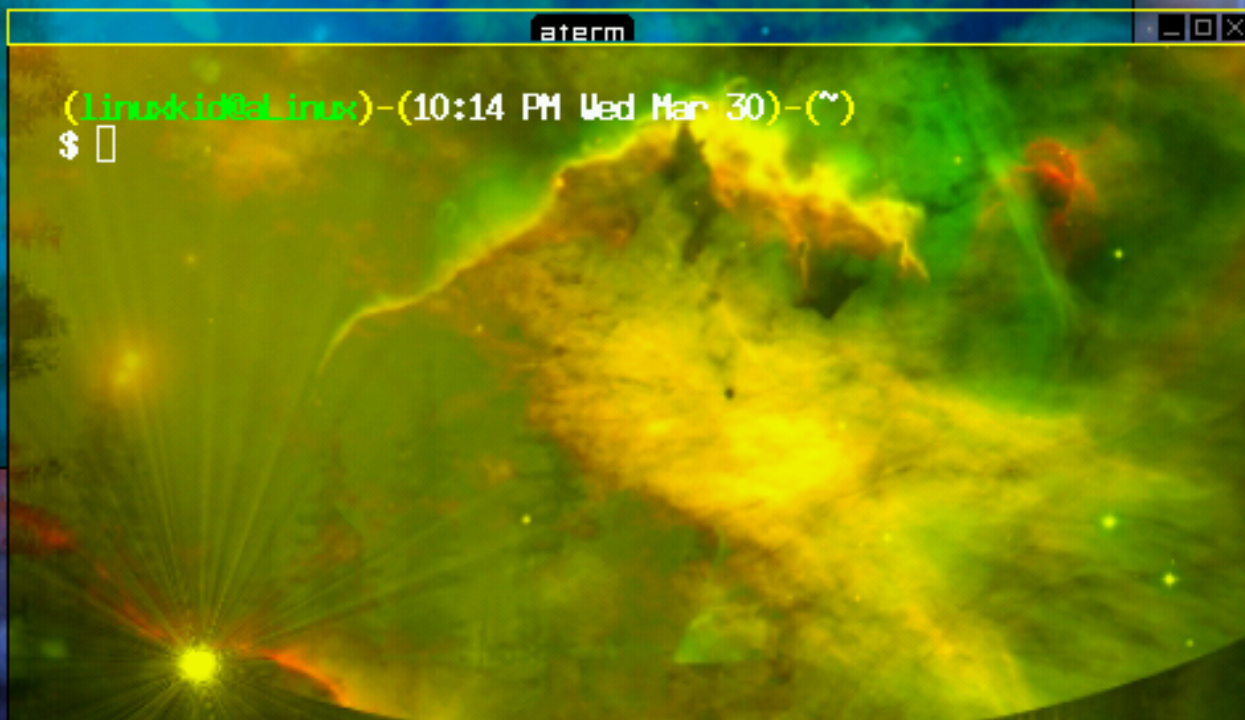
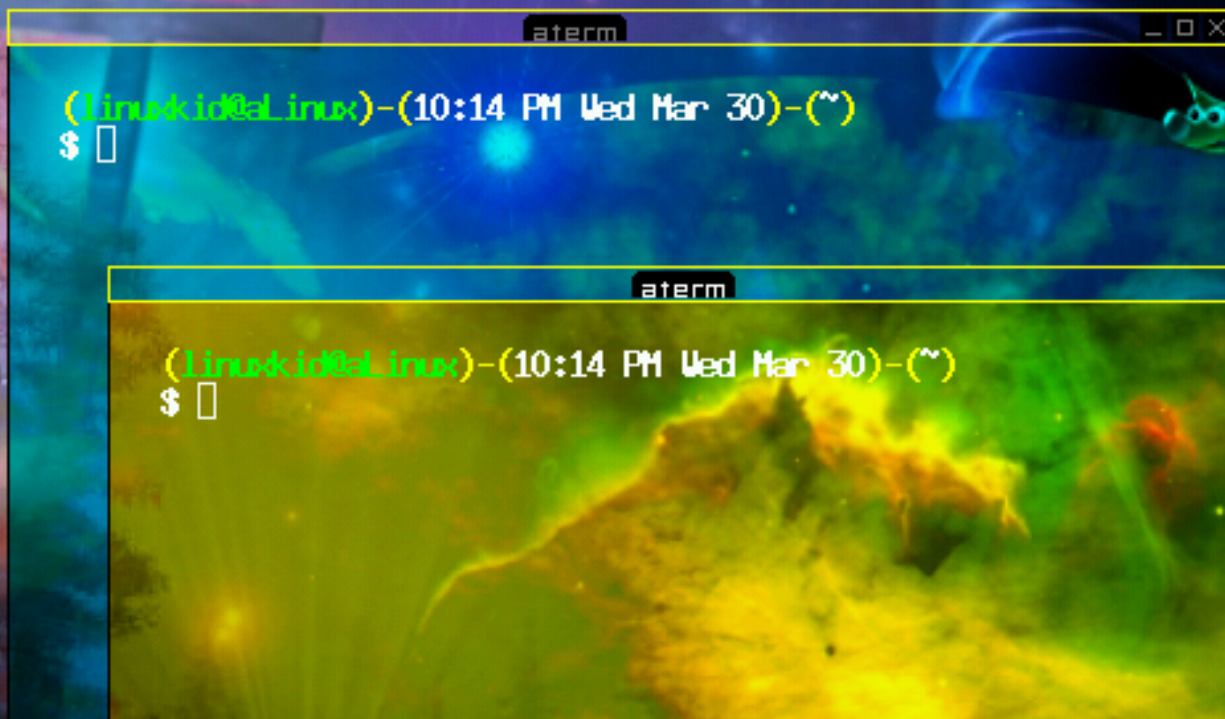


WAVC 02:51
TMP:46.0°F
DEW:36.0°F
PRS:30.18
HUM:67%
WIND:SE10.4

Uptime
1 days
1 hours
31 mins



WWW FTP GET



JUST LIKE BLADE RUNNER

...



X.Org



we modularised the build

(yay!)



but a bit too much

(boo!)



345 git modules — oops



but mostly we deleted stuff



xserver 1.0.2: 879,403 lines of code



xserver 1.0.2: 879,403 lines of code

xserver now: 562,678 lines of code



my net contribution:

$$\begin{array}{r} + 83,336 \\ - 186,985 \end{array}$$



eventually ran out of stuff to delete



another drawing model: XRender



four input stacks

core X11

X11 1.0, 2.0 & 2.2



sidenote:

three people on this earth
understand X input



really wish I wasn't one of them



five display management extensions

core X11

~~Xinerama~~

RandR 1.0, 1.2 & 1.4



four buffer management models

~~core X11~~

~~DRI~~

SHM

DRI2



hey internet peanut gallery



SHM and DRI2

don't work over the network



X isn't network-transparent.



we shifted all the paradigms



themes got harder

we drew them client-side



fonts got harder

we drew them client-side



subwindows got in the way

we moved them client-side



window management got harder
we got the WM to draw everything

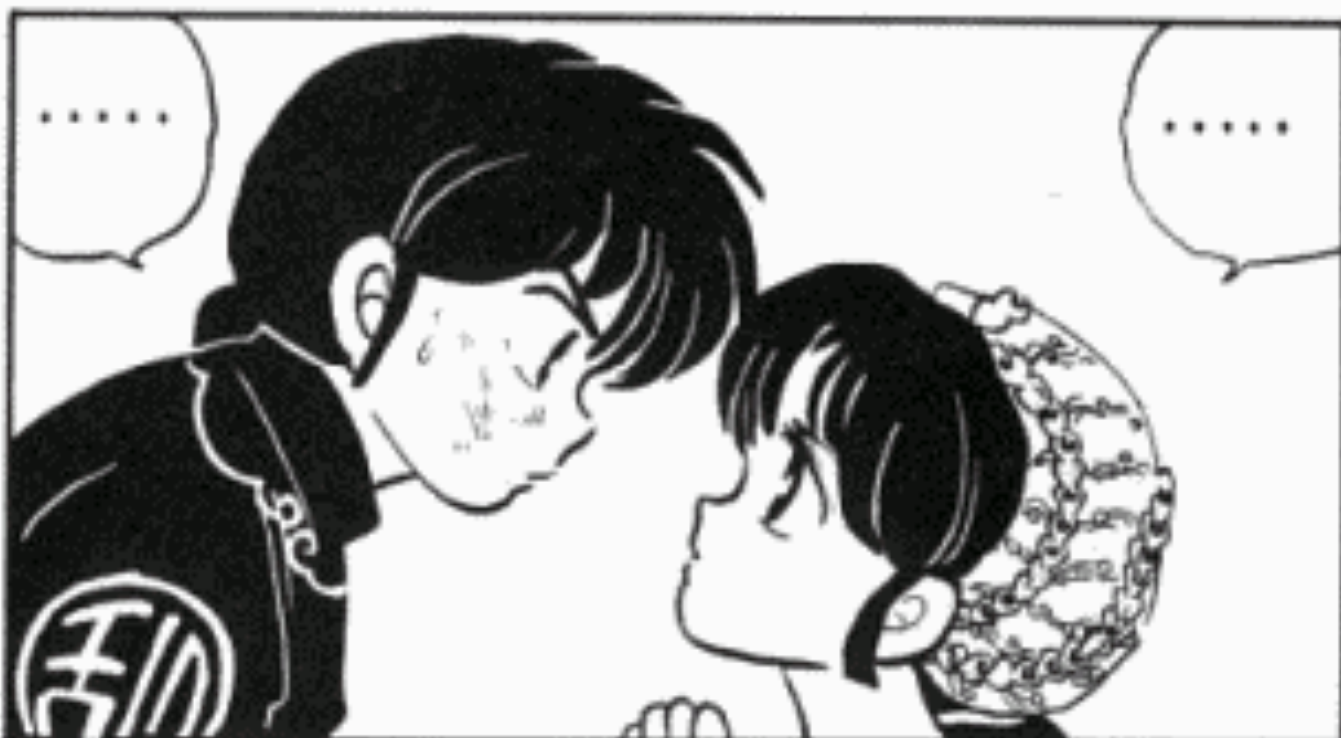
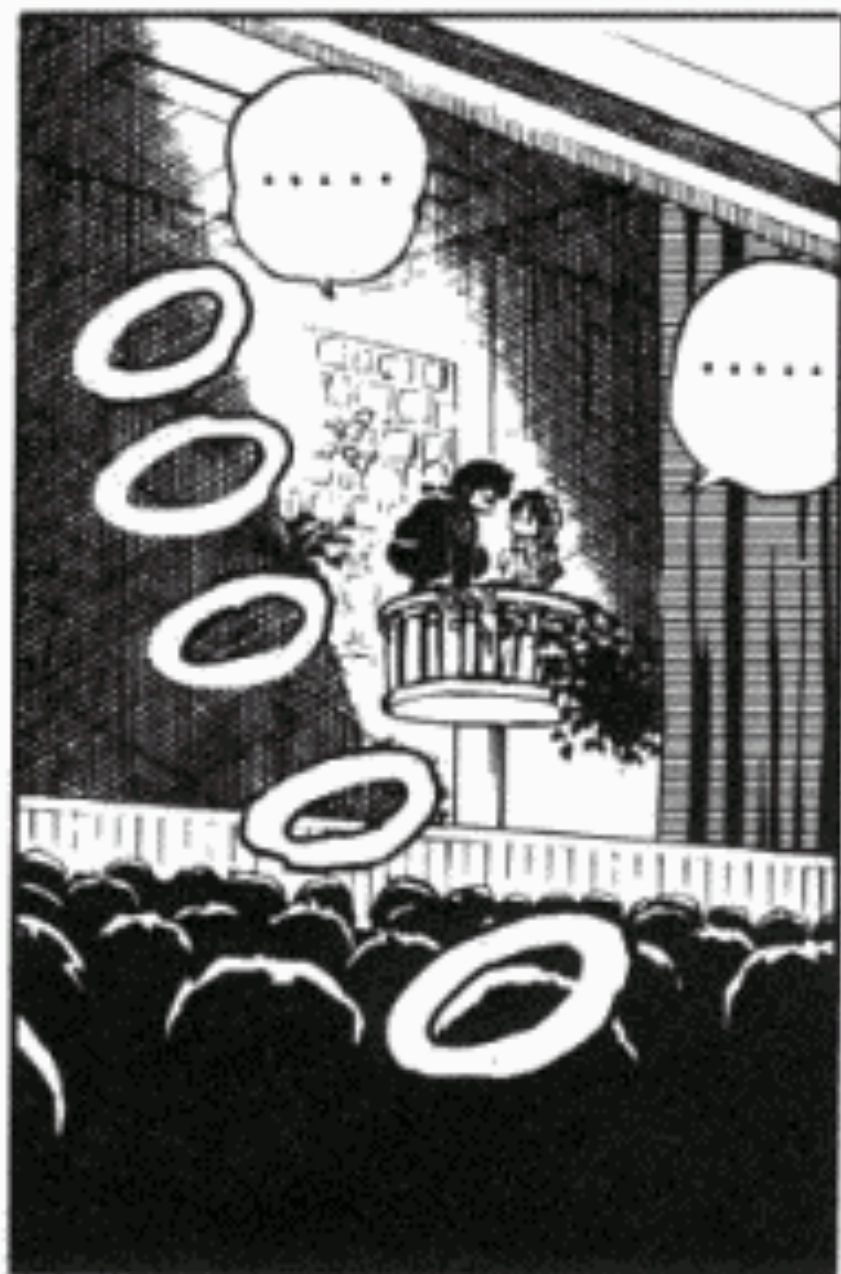


so what is the X server doing?

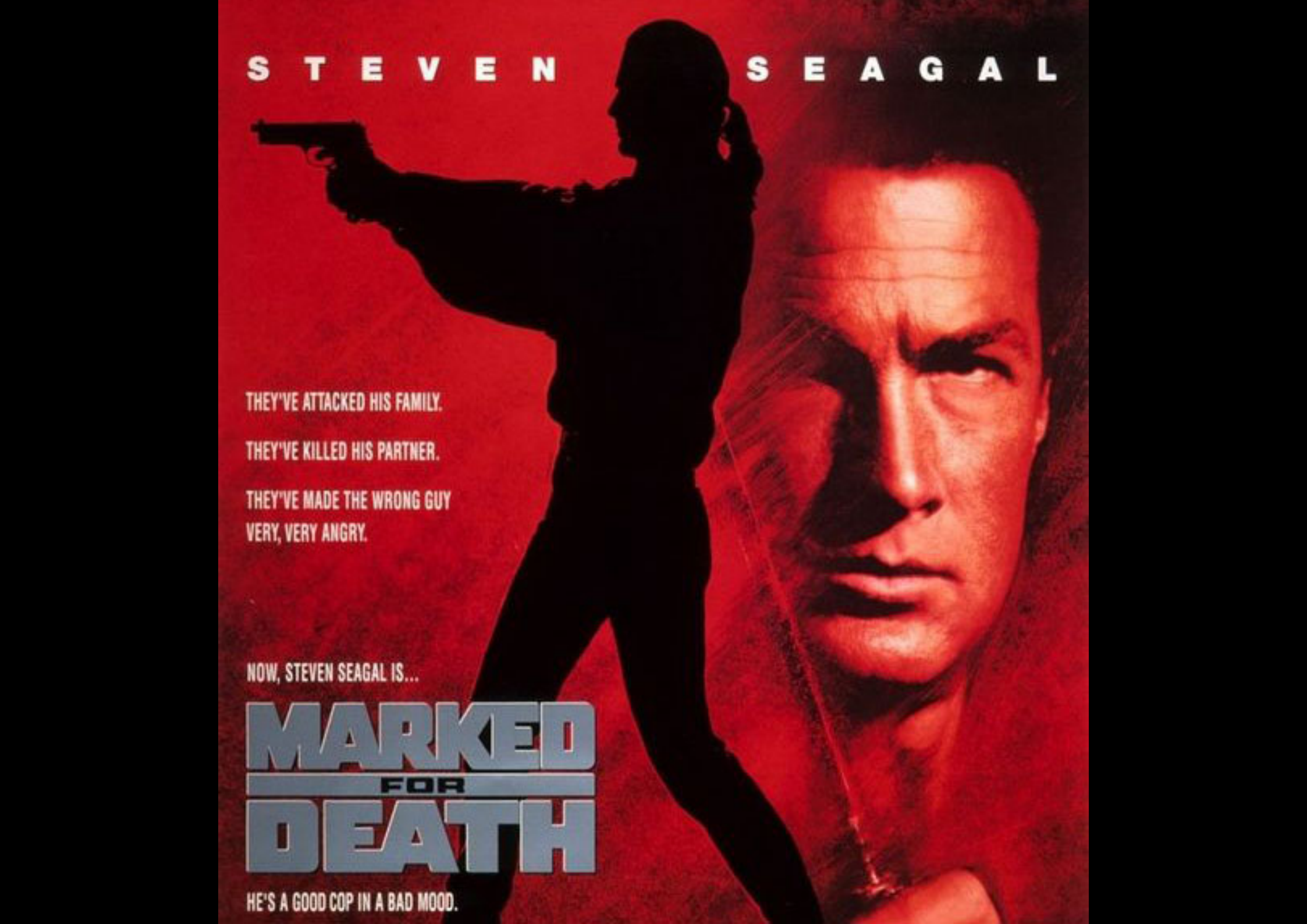


[dramatic pause]









S T E V E N S E A G A L

THEY'VE ATTACKED HIS FAMILY.

THEY'VE KILLED HIS PARTNER.

THEY'VE MADE THE WRONG GUY
VERY, VERY ANGRY.

NOW, STEVEN SEAGAL IS...

MARKED FOR DEATH

HE'S A GOOD COP IN A BAD MOOD.

**Narrate own dramatic
pause**



Defeat dramatic pause

...



not a lot.



clients render locally



clients tell the X server what they've drawn



the server asks the WM to display it



the window manager decides what to draw
and where



the X server displays what the WM rendered



the window manager is the new X server



and what's the X server?



really bad IPC



why really bad?



gedit startup:

130 blocking InternAtom calls

34 blocking GetProperty calls

116 property change requests



usually spends 25ms waiting for requests



usually spends 25ms waiting for requests

sometimes spends 1428ms



why?



when you resize
the server will draw on the window



this usually involves waiting for the GPU



which can take ages



when it's finished, the client will draw
over it anyway



so it's latent because it's
uselessly drawing the wrong thing



can we do better?



thought experiment:
a radically different model



clients render locally



clients tell the server what they've drawn



the server decides what to draw
and where



we just cut out the middle man



that's Wayland, in four slides.



XI I took about 90



hey internet peanut gallery



you say:

'X is totally The UNIX Way.'



UNIX says:

'Do one thing and do it well.'



what gives?



and while we're at it



'Those who do not understand UNIX
are condemned to quote Henry Spencer.'

— Kristian Høgsberg



right, so, Wayland



it's tractable



'every frame is perfect'



what's a frame?



a set of pixels that should be shown
in a window coherently at one time



remember X11 is immediate-mode



'draw a rectangle here'
'blit this image'
'render this text'



there's no boundary between them they'll
be displayed

at random times



you'll see incom

s



you'll see incomplete results



DRI2 almost fixes this, but not really



Wayland is solely frame-based



client says, 'display this'



server displays it



job done.



'every frame is perfect'



what's 'perfect'?



no flicker

no flashes

no tearing



ever.



it makes us look like amateurs.



descriptive, not prescriptive



pop-up window in X11:

give me all keyboard input

give me all mouse input

put this window exactly here



screensaver in X11:

give me all keyboard input

give me all mouse input

put this window fullscreen, topmost



result:

can't use volume keys during pop-up

can't use volume keys during screensaver

screensaver won't trigger during pop-up



it's been broken for 26 years

we tried to fix it

we can't.



IT'S 2013.

THIS IS NOT OK.



pop-up window in Wayland:

'this window is a pop-up window,
triggered from this click.
do the right thing, please'



it's all up to the compositor



gee but that sounds complex!



so are window managers.



yes, you can write bad composers



don't run them.



oh, and the screensaver



it's part of the compositor



random clients don't take your password



event driven



listening for input devices in X11



register for device notifications

ask for device list

wait for reply

parse reply

parse notifications, if any



listening for input devices in Wayland



register for device notifications

parse notifications, if any



dynamic everything by default.



proper object lifetimes



X Error: BadDevice (invalid input device)
Major opcode: 144
Minor opcode: 19



familiar?



objects in XLI can disappear at any time

the error is fatal by default



Wayland object destruction is client-side



listeners vs. clients



everything not global in X is per-client



e.g. input behaviour changes depending on
the version of the client's input extension



fine, right?



modern browser has four components:

browser frontend

toolkit used by backend

browser backend

plugins used by backend



your frontend supports Xi 2.1

your toolkit supports Xi 2.3

your backend supports Xi 2.0

your plugins only support core X11



what happens now?



your guess is as good as mine.



in Wayland, they all register listeners



listeners are versioned



everyone gets exactly what they want



sounds great, but can i use it?







LONDON BOROUGH OF
**WESTON
RISE**
W.C.1

ISLINGTON

Weston

reference compositor

straightforward plugin system

external 'shells' for WM/dock/etc

supports X11 clients



output: KMS, fbdev, Raspberry Pi
rendering: GLES or Pixman (software)
and hardware video overlays



QMLCompositor

easily write a compositor in QML

has too much C++ for my liking

but it does work



Mutter

has an out-of-date port

hybrid X11/Wayland compositor

GNOME Shell was running on it



GTK+, Clutter, Qt

ports all in upstream git

GStreamer

waylandvideosink exists, needs work



oh, and just one more thing



remember the bit about IPC?



guess what its worst case is ...



remoting!



can we do better?



kill the roundtrips: local compositor



easy on the bandwidth: image compression



hey presto, we're now on par with VNC



we've been experimenting already



experimental branch for Weston already up



we think it will be better than X



we think it can't be worse than X



and on that bombshell ...



~~subliminal message:~~

~~<http://www.collabora.com>~~

