



Presenting

Developing Device Drivers in Rust

School of Computing, Engineering & Physical Sciences

BSc (Honours) Computing Science

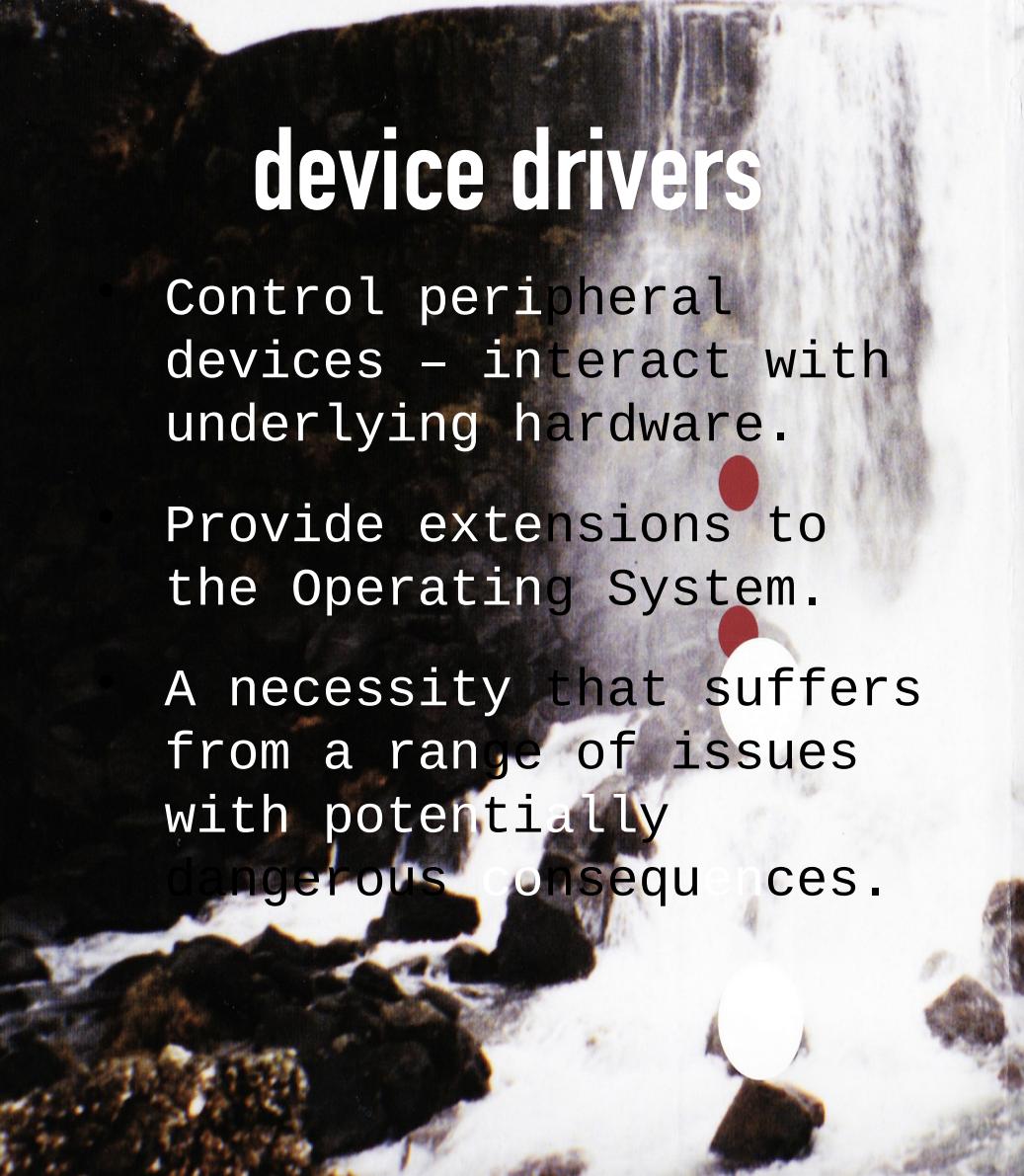
Supervisor: Paul Keir

Moderator: Stephen Devine

A close-up photograph of several orange flowers, possibly snapdragons, with long green stems. The flowers are in various stages of bloom, from tight buds to fully open blossoms. The background is a soft, out-of-focus light blue or white, creating a dreamlike atmosphere.

device drivers

t h e f r a g i l e



device drivers

- Control peripheral devices - interact with underlying hardware.

- Provide extensions to the Operating System.

- A necessity that suffers from a range of issues with potentially dangerous consequences.

problems

- Continue to be written in C.
 - Originally developed 1969-1973.
 - Suffers from issues with memory safety.
- Memory safety can lead to critical vulnerabilities - mostly present in C, C++ and Assembly

project aim

Overcome previously described issues by developing a Linux device driver in r u s t.

rust for Linux

2019, Miguel Ojeda

introduce a new system programming language into
Linux kernel

memory safe language

- strong compiler
- borrow system
- variable lifetimes

rust

Stroustrup's criticism

"every safe language, including rust, has
loopholes allowing unsafe code"

memory safety

Accessing memory typically outside the bounds of a data structure which then provides a vector to attack from to conduct further exploitation/attack.

Android	>65% of High & Critical security bugs
Android (bluetooth & media components)	90% of vulnerabilities
IOS 12	66.3% of all vulnerabilities
MacOS Mojave	71.5% of all vulnerabilities
Chrome	~70% of serious security bugs
Microsoft	~70% of CVE vulnerabilities
Firefox (CSS subsystem)	73.9% of bugs
Ubuntu kernel	65% of CVEs (In security updates between November and May 2020)

(Alex Gaynor, 2020)

Each statistic is that of a large code base containing millions of lines of code.

All are written in C or C++.

Includes;

- Use-after-free
- Double-free
- Heap Buffer overflow
- Integer overflow
- Out-of-bounds read
- Out-of-bounds write



Image: David Carson

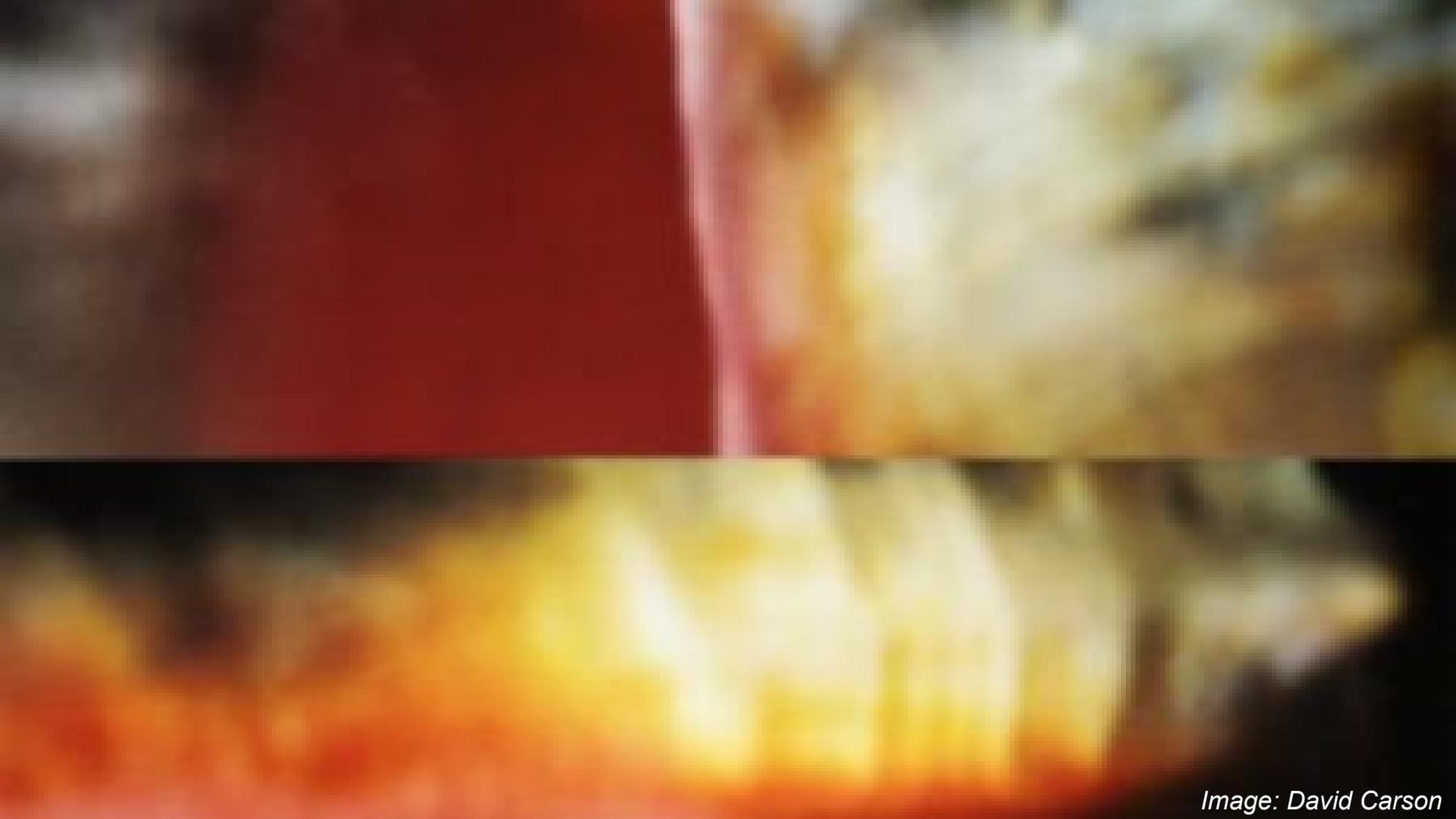
A blurry, abstract photograph of autumn foliage. The image is dominated by warm colors, including shades of red, orange, and yellow, which are typical of fall leaves. The background is out of focus, creating a soft, dreamlike quality. The overall composition is a wide, horizontal shot of a forest or garden during the autumn season.

Image: David Carson



Image: David Carson

the great below

(progress)

Industry input has informed development and other aspects of project.

- Alex Gaynor
- Miguel Ojeda
- Jonathan Blow

(development)

- Have enabled Rust support on Linux 6.1 Machine.
- USB support is under development within RFL.
- Not all kernel subsystems are implemented.
- Conclusion seems to be the that the project is still quite young but has a bright future.
- Demo: compilation & execution of 'Hello, World' in a Rust driver.

(findings)

Google & Android 13

Significant drop in memory safety vulns & severity.

Now 35% of total Android vulns (previously 76%).

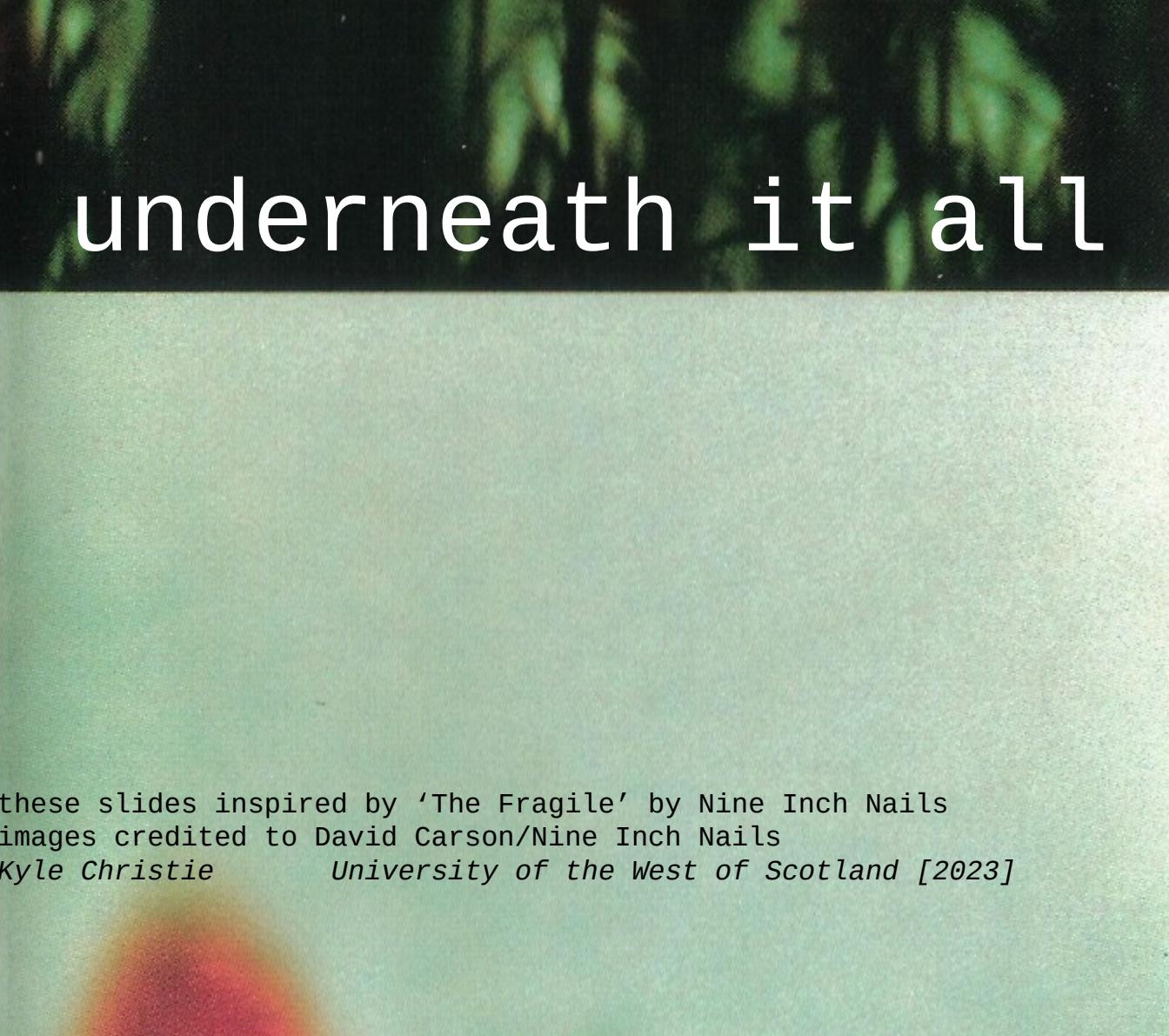
Exo-kernel

Remove as many hardware abstractions as possible.

Allow the application to control its own memory resources.

Paging, Scheduling, Context, Faults.

Improve performance, efficiency, development, testing.



underneath it all

these slides inspired by 'The Fragile' by Nine Inch Nails
images credited to David Carson/Nine Inch Nails
Kyle Christie *University of the West of Scotland [2023]*