Clueless OS Development

Daniel Maslowski
Agenda

How it started
How it’s going
Interlude: RISC-V
Future OS development
How it started
Little did I know…

Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98. We had an Internet connection via modem, later ISDN, then DSL. We had a Linux club at school. I wasn’t interested.

As a schoolmate had me checkout Fedora Core 1. I dumped it soon.

Happenstance

When I had my own apartment, I assembled a PC, running Windows XP. At some point, I somehow found something called Oberon. Mysterious. It was a tiny OS that fit on a single 1.4MB floppy disk. I tried it. I had no idea how to use it, but I had a mouse cursor on the screen.

And I had a NAS running some vendor’s hacked up Linux-based OS. That didn’t gain much of my interest, was just supposed to work.
Little did I know…

Early days
Little did I know…

Early days
Yea, I started with Windows… 3.11 was on my father’s PC.
Little did I know…

Early days
Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday.
Little did I know…

**Early days**

Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98.
Little did I know…

**Early days**
Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98. And we got Internet access via modem, later ISDN, then DSL.
Little did I know…

**Early days**

Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98. And we got Internet access via modem, later ISDN, then DSL. We had a Linux club at school. I wasn’t interested.
Little did I know…

Early days
Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98. And we got Internet access via modem, later ISDN, then DSL. We had a Linux club at school. I wasn’t interested. A school mate had me check out Fedora Core 1. I dumped it soon.
Little did I know…

Early days
Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98. And we got Internet access via modem, later ISDN, then DSL. We had a Linux club at school. I wasn’t interested. A school mate had me check out Fedora Core 1. I dumped it soon.

Happenstance
Little did I know…

Early days
Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98. And we got Internet access via modem, later ISDN, then DSL. We had a Linux club at school. I wasn’t interested. A school mate had me check out Fedora Core 1. I dumped it soon.

Happenstance
When I had my own apartment, I assembled a PC, running Windows XP.
Little did I know…

**Early days**
Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98. And we got Internet access via modem, later ISDN, then DSL. We had a Linux club at school. I wasn’t interested. A school mate had me check out Fedora Core 1. I dumped it soon.

**Happenstance**
When I had my own apartment, I assembled a PC, running Windows XP. At some point, I somehow found something called Oberon. Mysterious.
Little did I know…

Early days
Yea, I started with Windows… 3.11 was on my father’s PC. 
At some point, my uncle gave me a C64 for my birthday. 
Later, I got my own PC, running Windows 98. 
And we got Internet access via modem, later ISDN, then DSL. 
We had a Linux club at school. I wasn’t interested. 
A school mate had me check out Fedora Core 1. I dumped it soon.

Happenstance
When I had my own apartment, I assembled a PC, running Windows XP. 
At some point, I somehow found something called Oberon. Mysterious. 
It was a tiny OS that fit on a single 1.4MB floppy disk. I tried it.
Little did I know…

**Early days**
Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98. And we got Internet access via modem, later ISDN, then DSL. We had a Linux club at school. I wasn’t interested. A school mate had me check out Fedora Core 1. I dumped it soon.

**Happenstance**
When I had my own apartment, I assembled a PC, running Windows XP. At some point, I somehow found something called Oberon. Mysterious. It was a tiny OS that fit on a single 1.4MB floppy disk. I tried it. I had no idea how to use it, but I had a mouse cursor on the screen.
Little did I know…

Early days
Yea, I started with Windows… 3.11 was on my father’s PC. At some point, my uncle gave me a C64 for my birthday. Later, I got my own PC, running Windows 98. And we got Internet access via modem, later ISDN, then DSL. We had a Linux club at school. I wasn’t interested. A school mate had me check out Fedora Core 1. I dumped it soon.

Happenstance
When I had my own apartment, I assembled a PC, running Windows XP. At some point, I somehow found something called Oberon. Mysterious. It was a tiny OS that fit on a single 1.4MB floppy disk. I tried it. I had no idea how to use it, but I had a mouse cursor on the screen. And I had a NAS running some vendor’s hacked up Linux based OS.
Little did I know…

**Early days**
Yea, I started with Windows… 3.11 was on my father’s PC.
At some point, my uncle gave me a C64 for my birthday.
Later, I got my own PC, running Windows 98.
And we got Internet access via modem, later ISDN, then DSL.
We had a Linux club at school. I wasn’t interested.
A school mate had me check out Fedora Core 1. I dumped it soon.

**Happenstance**
When I had my own apartment, I assembled a PC, running Windows XP.
At some point, I somehow found something called Oberon. Mysterious.
It was a tiny OS that fit on a single 1.4MB floppy disk. I tried it.
I had no idea how to use it, but I had a mouse cursor on the screen.
And I had a NAS running some vendor’s hacked up Linux based OS.
That didn’t gain much of my interest, was just supposed to work.
Welcome to the Oberon page of the ETH Zurich

Welcome to the ETH Oberon web site of the Native Systems Group located at the Computer Systems Institute, Department of Computer Science. ETH (Swiss Federal Institute of Technology). Oberon is the name of a programming language in the Pascal/Modula tradition. Originally 'Oberon' was also the name of the runtime systems. For practical reasons the systems nomenclature was changed, while several new systems based on the Oberon language were developed by young generations of computer scientists at ETH (AOS (2003), Bluebottle(2005), A2 (2008))

The new System since 2008 is now called A2.
A2 is the name of a modern integrated software environment. It is a single-user, multi-core, multi-tasking system that runs on bare hardware or on top of a host operating system.

The newest developments of the Oberon Language and the A2 System as well as specific optimized system applications form the core of current research by the ETH Native Systems Group, lead by Prof. Jürg Gutknecht.

The Oberon project was launched in 1985 by Niklaus Wirth and Jürg Gutknecht at ETH. Although the project was originally targeted towards in-house hardware, the language and system have now been ported to many computer platforms.

Details in ETH Oberon Download Information.

The Oberon system is available free of charge and no registration is required for downloading the material. The source code is available under the following license agreement.

© 2008 ETH Zurich | Imprint | Disclaimer | 10 July 2008

At uni

I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the Sage math environment. Ubuntu broke with libraries I needed from the testing repos. So my friend recommended Gentoo instead. That really worked well.

Later I tried some other distros, and at some point discovered BSD. I played with GhostBSD, started listening to BSD Now from a podcast. I installed OpenBSD, and I think it by accident at openSUSE Conference. And I attended lots of other conferences, including EuroBSDCon.

Conferences are a great way to get knowledge and into communities.
At uni

Linux again
At uni

Linux again

I studied IT Security, and a friend of mine said I should look into Linux.
At uni

Linux again
I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with.
At uni

Linux again
I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the SAGE math environment.
At uni

**Linux again**

I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the SAGE math environment. Ubuntu broke with libraries I needed from the testing repos.
At uni

Linux again
I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the SAGE math environment. Ubuntu broke with libraries I needed from the testing repos. So my friend recommended Gentoo instead. That really worked well.
At uni

Linux again
I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the SAGE math environment. Ubuntu broke with libraries I needed from the testing repos. So my friend recommended Gentoo instead. That really worked well.

Some BSDs
At uni

Linux again
I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the SAGE math environment. Ubuntu broke with libraries I needed from the testing repos. So my friend recommended Gentoo instead. That really worked well.

Some BSDs
Later I tried some other distros, and at some point discovered BSD.
At uni

Linux again
I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the SAGE math environment. Ubuntu broke with libraries I needed from the testing repos. So my friend recommended Gentoo instead. That really worked well.

Some BSDs
Later I tried some other distros, and at some point discovered BSD. I played with GhostBSD, started listening to BSDNow from ca episode 1.
At uni

Linux again
I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the SAGE math environment. Ubuntu broke with libraries I needed from the testing repos. So my friend recommended Gentoo instead. That really worked well.

Some BSDs
Later I tried some other distros, and at some point discovered BSD. I played with GhostBSD, started listening to BSDNow from ca episode 1. I installed OpenBSD, and shred it by accident at openSUSE Conference.
Linux again
I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the SAGE math environment. Ubuntu broke with libraries I needed from the testing repos. So my friend recommended Gentoo instead. That really worked well.

Some BSDs
Later I tried some other distros, and at some point discovered BSD. I played with GhostBSD, started listening to BSDNow from ca episode 1. I installed OpenBSD, and shredded it by accident at openSUSE Conference. And I attended lots of other conferences, including EuroBSDCon.
At uni

Linux again
I studied IT Security, and a friend of mine said I should look into Linux. He suggested Ubuntu, said it was easy to begin with. We had to do some homework in the SAGE math environment. Ubuntu broke with libraries I needed from the testing repos. So my friend recommended Gentoo instead. That really worked well.

Some BSDs
Later I tried some other distros, and at some point discovered BSD. I played with GhostBSD, started listening to BSDNow from ca episode 1. I installed OpenBSD, and shred it by accident at openSUSE Conference. And I attended lots of other conferences, including EuroBSDCon. Conferences are a great way to get knowledge and into communities.
How it’s going
Leveraging knowledge

Building software

At university, I learned some C, some Java, some Atmel assembly. Through Linux and Gentoo especially, I learned about toolchains. Now I could build software and write small programs. Hello world…

Curiosity

My day job is web development, where new things happen all the time. I still keep my eyes open for operating systems here and there. What operating systems have emerged over the years?
Leveraging knowledge

Building software

At university, I learned some C, some Java, some Atmel assembly. Through Linux and Gentoo especially, I learned about toolchains. Now I could build software and write small programs. Hello world…

Curiosity

My day job is web development, where new things happen all the time. I still keep my eyes open for operating systems here and there. What operating systems have emerged over the years?
Leveraging knowledge

Building software
At university, I learned some C, some Java, some Atmel assembly.
Leveraging knowledge

Building software
At university, I learned some C, some Java, some Atmel assembly. Through Linux and Gentoo especially, I learned about toolchains.
Leveraging knowledge

Building software
At university, I learned some C, some Java, some Atmel assembly. Through Linux and Gentoo especially, I learned about toolchains. Now I could build software and write small programs. Hello world…
Leveraging knowledge

Building software
At university, I learned some C, some Java, some Atmel assembly. Through Linux and Gentoo especially, I learned about toolchains. Now I could build software and write small programs. Hello world…

Curiosity
My dayjob is web development, where new things happen all the time.
Leveraging knowledge

Building software
At university, I learned some C, some Java, some Atmel assembly. Through Linux and Gentoo especially, I learned about toolchains. Now I could build software and write small programs. Hello world…

Curiosity
My dayjob is web development, where new things happen all the time. I still keep my eyes open for operating systems here and there.
Leveraging knowledge

Building software
At university, I learned some C, some Java, some Atmel assembly. Through Linux and Gentoo especially, I learned about toolchains. Now I could build software and write small programs. Hello world…

Curiosity
My dayjob is web development, where new things happen all the time. I still keep my eyes open for operating systems here and there. What operating systems have emerged over the years?
Plan 9

The heck is Plan 9… history?! It is really hard to find information because many references got lost. I did my research and found HarveyOS, Jehanne, 9legacy, 9front…

One evening at Chaosdorf, I cloned HarveyOS and followed instructions. I could compile it with Clang on FreeBSD and run it in QEMU.
Plan 9
The heck is Plan 9… history?!
Plan 9

The heck is Plan 9… history?!

It is really hard to find information because many references got lost.
Plan 9

The heck is Plan 9… history?!

It is really hard to find information because many references got lost.

I did my research and found Harvey OS, Jehanne, 9legacy, 9front…
Plan 9

The heck is Plan 9… history?!

It is really hard to find information because many references got lost.

I did my research and found Harvey OS, Jehanne, 9legacy, 9front…

One evening at Chaosdorf, I cloned Harvey OS and followed instructions.
Plan 9

The heck is Plan 9… history?!

It is really hard to find information because many references got lost.

I did my research and found Harvey OS, Jehanne, 9legacy, 9front…

One evening at Chaosdorf, I cloned Harvey OS and followed instructions.

I could compile it with Clang on FreeBSD and run it in QEMU.
Plan 9

The heck is Plan 9… history?!

It is really hard to find information because many references got lost.

I did my research and found Harvey OS, Jehanne, 9legacy, 9front…

One evening at Chaosdorf, I cloned Harvey OS and followed instructions.

I could compile it with Clang on FreeBSD and run it in QEMU.
Well, it had some bugs.
Let’s get real

I wanted to run Harvey on a real machine, so I created a USB drive. It did boot up to some degree. I saw some output on the screen. One message was this familiar line that always made me smile:

Hello, I am Harvey :-)

But it got stuck and did not make it into the graphical desktop.
Let’s get real

I wanted to run Harvey on a real machine, so I created a USB drive.
Let’s get real

I wanted to run Harvey on a real machine, so I created a USB drive.

It did boot up to some degree. I saw some output on the screen.
Let’s get real

I wanted to run Harvey on a real machine, so I created a USB drive.

It did boot up to some degree. I saw some output on the screen.

One message was this familiar line that always made me smile:

Hello, I am Harvey :-)
Let’s get real
I wanted to run Harvey on a real machine, so I created a USB drive.
It did boot up to some degree. I saw some output on the screen.
One message was this familiar line that always made me smile:
Hello, I am Harvey :-) 
But it got stuck and did not make it into the graphical desktop.
Destiny?

When I joined the Harvey Slack team (now Matrix), I met a friend.
Destiny?

When I joined the Harvey Slack team (now Matrix), I met a friend.

Was that coincidence, or was it destiny?!
Destiny?

When I joined the Harvey Slack team (now Matrix), I met a friend.

Was that coincidence, or was it destiny?!

I ended up debugging an Intel ethernet driver with JavaScript.

```javascript
const fs = require('fs');
const file = fs.readFileSync('./flashregion_3_gbe.bin');

let sum = 0;
for (let w = 0; w < 0x40; w++) {
    const d = file[w*2] + (file[w*2+1] << 8);
    console.info(w, d.toString(10)); // debug the heck out of it
    sum = (sum + d) & 0x10000;
}
console.info(sum.toString(16));
```
Interlude: RISC-V
We’re gonna need another ISA!

---

2 https://en.wikipedia.org/wiki/Instruction_set_architecture
We’re gonna need another ISA!

An *Instruction Set Architecture* is an abstract model of a computer.²

²https://en.wikipedia.org/wiki/Instruction_set_architecture
We’re gonna need another ISA!

An *Instruction Set Architecture* is an abstract model of a computer.\(^2\)

Through the oreboot firmware project, I ran into RISC-V.

\(^2\)https://en.wikipedia.org/wiki/Instruction_set_architecture
We’re gonna need another ISA!

An *Instruction Set Architecture* is an abstract model of a computer.\(^2\)

Through the oreboot firmware project, I ran into RISC-V.

And I got to know Michael Engel, who invited me to Bamberg.

\(^2\)https://en.wikipedia.org/wiki/Instruction_set_architecture
We’re gonna need another ISA!

An *Instruction Set Architecture* is an abstract model of a computer.\(^2\)

Through the oreboot firmware project, I ran into RISC-V.

And I got to know Michael Engel, who invited me to Bamberg.

TL;DR that’s why I’m here today. :)

\(^2\)https://en.wikipedia.org/wiki/Instruction_set_architecture
Future OS development
Destination IWP9
Destination IWP9

After a decade, it was time for another International Workshop on Plan 9.
Destination IWP9

After a decade, it was time for another International Workshop on Plan 9.

The odds: 9th IWP9, in the year of the rabbit. Happenstance?
Destination IWP9

After a decade, it was time for another International Workshop on Plan 9.

The odds: 9th IWP9, in the year of the rabbit. Happenstance?

We met at Uni Waterloo, Canada, and spent three whole days together.
What is an OS now anyway?

An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.

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

OSE (Operating Systems Engineering) was taught there at Universität Bamberg by Prof. Michael Engel.

- Data structures (just like all other software)
- Memory/resourcemanagement (this is crucial and hard)
- Processes and scheduling (can be very simple)
- Inter-process communication (where it gets nasty)
- Files and filesystems (well, not always necessary)
- Drivers (and lots of them for all the peripherals)
- Wonky, weird, wacko hardware workarounds (spectre/meltdown…)

https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
What is an OS now anyway?

An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.

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

\[3\]https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
What is an OS now anyway?

An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.

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

OSE (Operating Systems Engineering)
is taught here at Universität Bamberg by Prof. Michael Engel

3 https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
What is an OS now anyway?

An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.

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

OSE (Operating Systems Engineering) is taught here at Universität Bamberg³ by Prof. Michael Engel

- data structures (just like all other software)

³https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
What is an OS now anyway?

An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.

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

OSE (Operating Systems Engineering)

is taught here at Universität Bamberg³ by Prof. Michael Engel

- data structures (just like all other software)
- memory/resource management (this is crucial and hard)

³https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
What is an OS now anyway?

An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.

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

OSE (Operating Systems Engineering) is taught here at Universität Bamberg by Prof. Michael Engel

- data structures (just like all other software)
- memory/resource management (this is crucial and hard)
- processes and scheduling (can be very simple)

---

³https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
What is an OS now anyway?

An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.

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

OSE (Operating Systems Engineering) is taught here at Universität Bamberg by Prof. Michael Engel

- data structures (just like all other software)
- memory/resource management (this is crucial and hard)
- processes and scheduling (can be very simple)
- inter-process communication (where it gets nasty)

---

³https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
What is an OS now anyway?

An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.

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

OSE (Operating Systems Engineering)
is taught here at Universität Bamberg by Prof. Michael Engel

- data structures (just like all other software)
- memory/resource management (this is crucial and hard)
- processes and scheduling (can be very simple)
- inter-process communication (where it gets nasty)
- files and file systems (well, not always necessary)

---

³https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
What is an OS now anyway?

An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.

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

OSE (Operating Systems Engineering)

is taught here at Universität Bamberg³ by Prof. Michael Engel

- data structures (just like all other software)
- memory/resource management (this is crucial and hard)
- processes and scheduling (can be very simple)
- inter-process communication (where it gets nasty)
- files and file systems (well, not always necessary)
- drivers (and lots of them for all the peripherals)

³https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
What is an OS now anyway?

*An operating system (OS) is system software that manages computer hardware and software resources, and provides common services for computer programs.*

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

**OSE (Operating Systems Engineering)**

is taught here at Universität Bamberg\(^3\) by Prof. Michael Engel

- data structures (just like all other software)
- memory/resource management (this is crucial and hard)
- processes and scheduling (can be very simple)
- inter-process communication (where it gets nasty)
- files and file systems (well, not always necessary)
- drivers (and lots of them for all the peripherals)
- wonky weird wacko hardware workarounds (spectre/meltdown…)

---

\(^3\)https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/
OSDI, oh my
OSDI, oh my

Operating Systems Design and Implementation

Related

Next SOSP (Symposium on Operating Systems Principles) will be in Koblenz, Germany:

https://sosp2023.mpi-sws.org/

Together with PLOS (Workshop on Programming Languages and Operating Systems):

https://plos-workshop.org/2023/

Topics

A wider range of topics is discussed at such conferences, such as

performance and efficiency

security and hardening

data persistence, transfer and transactions

shift from OS/PL co-design to PL ideas in influencing OS design

---

4 https://dl.acm.org/conference/osdi

5 https://www.usenix.org/conference/osdi23
OSDI, oh my

Operating Systems Design and Implementation
A field of study in a way, a conference theme at both ACM⁴ and Usenix⁵.

---

⁴https://dl.acm.org/conference/osdi
⁵https://www.usenix.org/conference/osdi23
OSDI, oh my

Operating Systems Design and Implementation
A field of study in a way, a conference theme at both ACM\(^4\) and Usenix\(^5\).

Related
Next SOSP (Symposium on Operating Systems Principles) will be in Koblenz, Germany: [https://sosp2023.mpi-sws.org/](https://sosp2023.mpi-sws.org/)
Together with PLOS (Workshop on Programming Languages and Operating Systems): [https://plos-workshop.org/2023/](https://plos-workshop.org/2023/)

\(^4\)https://dl.acm.org/conference/osdi
\(^5\)https://www.usenix.org/conference/osdi23
OSDI, oh my

Operating Systems Design and Implementation
A field of study in a way, a conference theme at both ACM⁴ and Usenix⁵.

Related
Next SOSP (Symposium on Operating Systems Principles) will be in Koblenz, Germany: https://sosp2023.mpi-sws.org/
Together with PLOS (Workshop on Programming Languages and Operating Systems): https://plos-workshop.org/2023/

Topics

---

⁴https://dl.acm.org/conference/osdi
⁵https://www.usenix.org/conference/osdi23
OSDI, oh my

Operating Systems Design and Implementation
A field of study in a way, a conference theme at both ACM\(^4\) and Usenix\(^5\).

Related
Next SOSP (Symposium on Operating Systems Principles) will be in Koblenz, Germany: https://sosp2023.mpi-sws.org/
Together with PLOS (Workshop on Programming Languages and Operating Systems): https://plos-workshop.org/2023/

Topics
A wide range of topics is discussed at such conferences, such as
- performance and efficiency
- security and hardening
- data persistence, transfer and transactions
- shift from OS/PL co-design to PL ideas influencing OS design

\(^4\)https://dl.acm.org/conference/osdi
\(^5\)https://www.usenix.org/conference/osdi23
Motty’s criticism
Mothy’s criticism

Timothy Roscoe
Systems Group of the Computer Science Department
at ETH Zurich
https://people.inf.ethz.ch/troscoe/

---

6 https://www.usenix.org/conference/osdi21/presentation/fri-keynote
Mothy’s criticism

Timothy Roscoe
Systems Group of the Computer Science Department
at ETH Zurich
https://people.inf.ethz.ch/troscoe/

It’s Time for Operating Systems to Rediscover Hardware\textsuperscript{6}

\begin{itemize}
  \item OSes not covering the entire platform (I agree)
\end{itemize}

\textsuperscript{6}https://www.usenix.org/conference/osdi21/presentation/fri-keynote
\textsuperscript{7}https://dl.acm.org/doi/10.1145/3593856.3595903
Mothy’s criticism

Timothy Roscoe
Systems Group of the Computer Science Department at ETH Zurich
https://people.inf.ethz.ch/troscoe/

It’s Time for Operating Systems to Rediscover Hardware⁶

- OSes not covering the entire platform (I agree)

Putting out the hardware dumpster fire⁷

- attempt to model platform components’ dependencies
- define trust relationships in the hardware system
- leverage the Rust programming language

---

⁶https://www.usenix.org/conference/osdi21/presentation/fri-keynote
⁷https://dl.acm.org/doi/10.1145/3593856.3595903
Redox

A Unix-like OS, inspired by Plan 9, Minix, BSD and Linux, written in Rust

https://www.redox-os.org/
Dream OSes
Dream OSes

IDROS (Instrumented Distributed Resources OS)
- everything is a resource on the network
- goal: no services; be self-serving

\[8\text{https://hostile.education/9loa/}\]
Dream OSes

IDROS (Instrumented Distributed Resources OS)
- everything is a resource on the network
- goal: no services; be self-serving

9loa
- an operating system sits between a human and a machine

---

8 https://hostile.education/9loa/
A new OS, inspired by Plan 9, written in Rust

https://github.com/r9os/r9
R9 DEMO
Small computers everywhere
Small computers everywhere

MCUs getting closer to application processors
Small computers everywhere

- MCU getting closer to application processors
- General purpose, special purpose, or real-time OS?
Small computers everywhere

- MCUs getting closer to application processors
- general purpose, special purpose, or real-time OS?
- FreeRTOS, Zephyr, Hubris, embOS, EPOS, LiteOS, Melis…

https://github.com/orangecms/ch32v307-rust
Small computers everywhere

- MCUs getting closer to application processors
- general purpose, special purpose, or real-time OS?
- FreeRTOS, Zephyr, Hubris, embOS, EPOS, LiteOS, Melis…
- Fuchsia, HarmonyOS, rCore, zCore, Theseus, Hermit etc
Small computers everywhere

- MCUs getting closer to application processors
- general purpose, special purpose, or real-time OS?
- FreeRTOS, Zephyr, Hubris, embOS, EPOS, LiteOS, Melis…
- Fuchsia, HarmonyOS, rCore, zCore, Theseus, Hermit etc

https://github.com/orangecms/ch32v307-rust
MnemOS

9 https://onevariable.com/blog/mnemos-moment-1/
MnemOS for projects that exist in the liminal space between “too big/complicated for bare metal or a simple RTOS”, and “too small/underpowered/time critical for Linux”\(^9\)

\(^9\)https://onevariable.com/blog/mnemos-moment-1/
MnemOS

for projects that exist in the liminal space between “too big/complicated for bare metal or a simple RTOS”, and “too small/underpowered/time critical for Linux”

https://github.com/tosc-rs/mnemos

---

9https://onevariable.com/blog/mnemos-moment-1/
Gadget hacking and development boards
Gadget hacking and development boards

Daniel Maslowski aka CyReVolt
@OrangeCMS

This S903X4 Android TV Box runs its UART at 921600 baud. The kernel is a 5.4.125 built with clang. It’s amazing how you can get a great dev board by just buying a mass product. :-)

Gadget hacking and development boards

Linux is “just a kernel”, but only needs a single command: /init

meet u-root and a Linux port of cpu
cpu DEMO
User Interface Design

Why is this not a topic in OSDI? Or... is it? It was, many years ago.

A Minimalist Global User Interface (1991)

Combo of editor + windows system + shell + user interface

Genuinely Functional User Interfaces (2001)

GUI library for Haskell based on formal model of user interfaces

Touch input

Contemporary devices commonly feature touchscreens. Sailfish OS explored gestures quite a lot.

How about a three-finger diagonal to draw an applauncher?


https://sailfishos.org/design/gestures/
User Interface Design

Why is this not a topic in OSDI? Or… is it? It was, many years ago.

---


---
User Interface Design

Why is this not a topic in OSDI? Or... is it? It was, many years ago.

A Minimalist Global User Interface\textsuperscript{10} (1991)
combo of editor + window system + shell + user interface

\textsuperscript{11}https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces
User Interface Design

Why is this not a topic in OSDI? Or… is it? It was, many years ago.

A Minimalist Global User Interface\(^{10}\) (1991)
combo of editor + window system + shell + user interface

Genuinely Functional User Interfaces\(^{11}\) (2001)
GUI library for Haskell based on formal model of user interfaces

\(^{11}\)https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces
User Interface Design

Why is this not a topic in OSDI? Or… is it? It was, many years ago.

A Minimalist Global User Interface\(^\text{10}\) (1991)
combo of editor + window system + shell + user interface

Genuinely Functional User Interfaces\(^\text{11}\) (2001)
GUI library for Haskell based on formal model of user interfaces

Touch input
Contemporary devices commonly feature touch screens.

\(^\text{11}\) https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces
\(^\text{12}\) https://sailfishos.org/design/gestures/
User Interface Design

Why is this not a topic in OSDI? Or… is it? It was, many years ago.

A Minimalist Global User Interface\(^{10}\) (1991)
combo of editor + window system + shell + user interface

Genuinely Functional User Interfaces\(^{11}\) (2001)
GUI library for Haskell based on formal model of user interfaces

Touch input
Contemporary devices commonly feature touch screens.
Sailfish OS explored gestures quite a lot.\(^{12}\)

\(^{11}\)https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces
\(^{12}\)https://sailfishos.org/design/gestures/
User Interface Design

Why is this not a topic in OSDI? Or… is it? It was, many years ago.

A Minimalist Global User Interface\(^{10}\) (1991)
combo of editor + window system + shell + user interface

Genuinely Functional User Interfaces\(^ {11}\) (2001)
GUI library for Haskell based on formal model of user interfaces

Touch input
Contemporary devices commonly feature touch screens.
Sailfish OS explored gestures quite a lot.\(^{12}\)
How about a three-finger diagonal to draw an app launcher?

\(^{10} \text{http://doc.cat-v.org/plan_9/1st_edition/help/help.pdf} \)
\(^{11} \text{https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces} \)
\(^{12} \text{https://sailfishos.org/design/gestures/} \)
Thank you! :)}
Related

Harvey OS - Glenda and the Gopher Rejoin (LNI 2020)

Drivers from Outer Space (CLT 2022)

IDROS - Instrumented Distributed Resources OS
https://metaspora.org/idros.pdf

Platform System Interface - Design und Evaluation holistischer Computerarchitektur (rC3 2022)
Follow Me

https://github.com/orangecms
https://twitter.com/orangecms
https://mastodon.social/@cyrevolt
https://youtube.com/@cyrevolt
https://twitch.tv/cyrevolt

Daniel Maslowski


License: CC BY 4.0 https://creativecommons.org/licenses/by/4.0/