





# 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...



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.



# Going back in time<sup>1</sup>

**ETH**

Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zürich

**OBERON**

News | About | People  
Language | Systems | Archives  
FAQ | Downloads | Bibliography | Partnership

Contact | Sitemap | Help

Search  GO

ETH Zurich - Oberon



## 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.

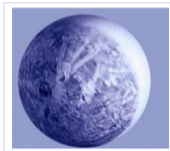
A2 is the name of a modern integrated *software environment*. It is a single-user, multi-core, multi-tasking system that runs on bare hardware or on top of a host operating system.

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

POWERED BY ZOPH AND SILVA - [ETH Web Office](#)

TOP



<sup>1</sup><https://web.archive.org/web/20080911231619/http://www.oberon.ethz.ch:80/>

At uni



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.



# 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.



# 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



# Leveraging knowledge

Building software



# 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



# 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

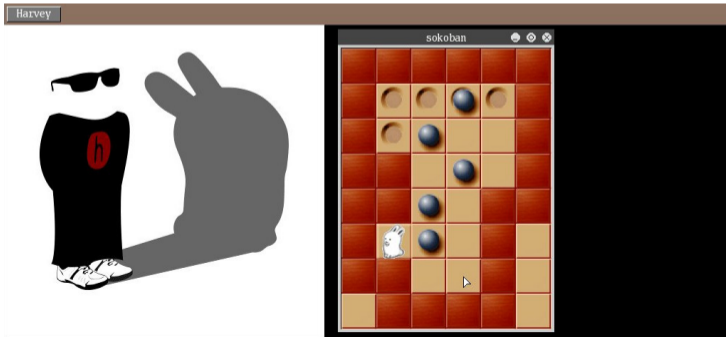
The heck is Plan 9... history?!

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.





# Boom!

Harvey

```
lock 0xffffffffb2c3f0 loop key 0xdeaddead pc 0xffffffff01254ad held by pc 0xffffffff0154f28 proc 1
8
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
lock 0xffffffffb2c3f0 loop key 0xdeaddead pc 0xffffffff01254ad held by pc 0xffffffff0154f28 proc 1
8
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
lock 0xffffffffb2c3f0 loop key 0xdeaddead pc 0xffffffff01254ad held by pc 0xffffffff0154f28 proc 1
8
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
lock 0xffffffffb2c3f0 loop key 0xdeaddead pc 0xffffffff01254ad held by pc 0xffffffff0154f28 proc 1
8
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
lock 0xffffffffb2c3f0 loop key 0xdeaddead pc 0xffffffff01254ad held by pc 0xffffffff0154f28 proc 1
8
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
dumpaproc: what are HSEG and BSEG 18:      jay pc 0x43cefc dbgpc 0x43cefc      pwrite (Running) ut 243
st 999 bss 0x0 qpc 0xffffffff014c7a0 nl 1 nd 0 lpc 0x1000000000 pri 10
```

Well, it had some bugs.



Let's get real



# 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

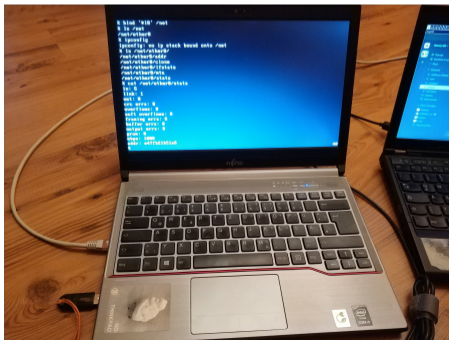
I wanted to run Harvey on a real machine, so I created a USB drive.

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.

```
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*s] + (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!

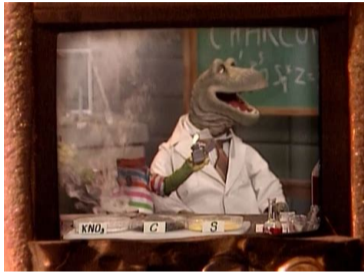


---

<sup>2</sup>[https://en.wikipedia.org/wiki/Instruction\\_set\\_architecture](https://en.wikipedia.org/wiki/Instruction_set_architecture)



We're gonna need another ISA!

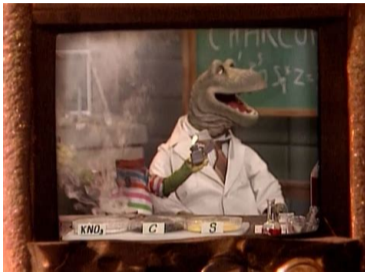


An *Instruction Set Architecture* is an abstract model of a computer.<sup>2</sup>

<sup>2</sup>[https://en.wikipedia.org/wiki/Instruction\\_set\\_architecture](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.<sup>2</sup>

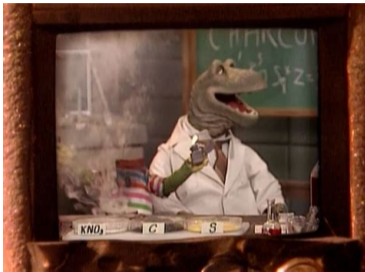
Through the oreboot firmware project, I ran into RISC-V.

---

<sup>2</sup>[https://en.wikipedia.org/wiki/Instruction\\_set\\_architecture](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.<sup>2</sup>

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

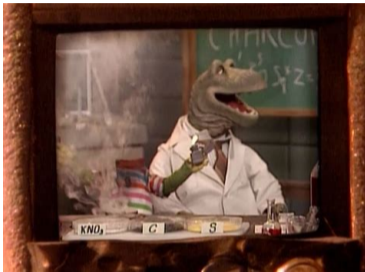
And I got to know Michael Engel, who invited me to Bamberg.

---

<sup>2</sup>[https://en.wikipedia.org/wiki/Instruction\\_set\\_architecture](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.<sup>2</sup>

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. :)

---

<sup>2</sup>[https://en.wikipedia.org/wiki/Instruction\\_set\\_architecture](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?

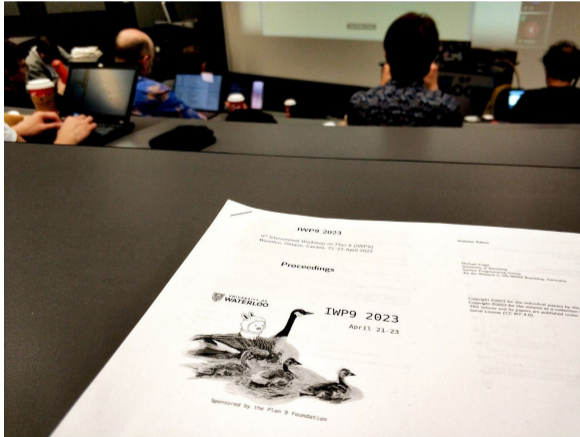


# Destination IWP9

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?



# 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](https://en.wikipedia.org/wiki/Operating_system)

---

<sup>3</sup><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](https://en.wikipedia.org/wiki/Operating_system)

OSE (Operating Systems Engineering)

is taught here at Universität Bamberg<sup>3</sup> by Prof. Michael Engel

---

<sup>3</sup><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](https://en.wikipedia.org/wiki/Operating_system)

OSE (Operating Systems Engineering)

is taught here at Universität Bamberg<sup>3</sup> by Prof. Michael Engel



data structures (just like all other software)

---

<sup>3</sup><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](https://en.wikipedia.org/wiki/Operating_system)

## OSE (Operating Systems Engineering)

is taught here at Universität Bamberg<sup>3</sup> by Prof. Michael Engel



data structures (just like all other software)



memory/resource management (this is crucial and hard)

---

<sup>3</sup><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](https://en.wikipedia.org/wiki/Operating_system)

## OSE (Operating Systems Engineering)

is taught here at Universität Bamberg<sup>3</sup> 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)

---

<sup>3</sup><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](https://en.wikipedia.org/wiki/Operating_system)

## OSE (Operating Systems Engineering)

is taught here at Universität Bamberg<sup>3</sup> 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)

---

<sup>3</sup><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](https://en.wikipedia.org/wiki/Operating_system)

## OSE (Operating Systems Engineering)

is taught here at Universität Bamberg<sup>3</sup> 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)

---

<sup>3</sup><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](https://en.wikipedia.org/wiki/Operating_system)

## OSE (Operating Systems Engineering)

is taught here at Universität Bamberg<sup>3</sup> 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)

---

<sup>3</sup><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](https://en.wikipedia.org/wiki/Operating_system)

## OSE (Operating Systems Engineering)

is taught here at Universität Bamberg<sup>3</sup> 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...)

---

<sup>3</sup><https://www.uni-bamberg.de/sysnap/studium/sommersemester/operating-systems-engineering/>



OSDI, oh my



OSDI, oh my

## Operating Systems Design and Implementation

---

<sup>4</sup><https://dl.acm.org/conference/osdi>

<sup>5</sup><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 ACM<sup>4</sup> and Usenix<sup>5</sup>.

---

<sup>4</sup><https://dl.acm.org/conference/osdi>

<sup>5</sup><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 ACM<sup>4</sup> and Usenix<sup>5</sup>.

### 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/>

---

<sup>4</sup><https://dl.acm.org/conference/osdi>

<sup>5</sup><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 ACM<sup>4</sup> and Usenix<sup>5</sup>.

### 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

---

<sup>4</sup><https://dl.acm.org/conference/osdi>

<sup>5</sup><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 ACM<sup>4</sup> and Usenix<sup>5</sup>.





### 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

---

<sup>4</sup><https://dl.acm.org/conference/osdi>

<sup>5</sup><https://www.usenix.org/conference/osdi23>



# Mothy's criticism

---



# Mothy's criticism



Timothy Roscoe

Systems Group of the Computer Science Department  
at ETH Zurich

<https://people.inf.ethz.ch/troscoe/>

---

<sup>6</sup><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<sup>6</sup>



OSes not covering the entire platform (I agree)

---

<sup>6</sup><https://www.usenix.org/conference/osdi21/presentation/fri-keynote>

<sup>7</sup><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 Hardware<sup>6</sup>



OSes not covering the entire platform (I agree)

## Putting out the hardware dumpster fire<sup>7</sup>



attempt to model platform components' dependencies



define trust relationships in the hardware system



leverage the Rust programming language

---

<sup>6</sup><https://www.usenix.org/conference/osdi21/presentation/fri-keynote>

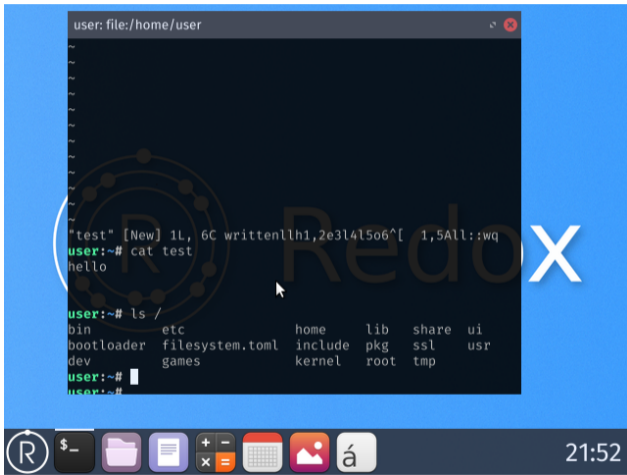
<sup>7</sup><https://dl.acm.org/doi/10.1145/3593856.3595903>





# 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



# Dream OSes

## IDROS (Instrumented Distributed Resources OS)

- 👤 everything is a resource on the network
- 👤 goal: no services; be self-serving

## 9loa<sup>8</sup>

- 👤 an operating system sits between a human and a machine



<sup>8</sup><https://hostile.education/9loa/>



## A new OS, inspired by Plan 9, written in Rust

```

dama@orangelemp ~/f/o/s/m/s/nezha (dl-nosbi)>
dama@orangelemp ~/P/r9 (mangopi) [1]> file ~/Projects/r9/target/riscv64-unknown-none-elf/debug/riscv64
/home/dama/Projects/r9/target/riscv64-unknown-none-elf/debug/riscv64: ELF 64-bit LSB executable, UCB RISC-V, RVC, double-float ABI, version 1 (SYSV), statically linked, with debug_info, not stripped
dama@orangelemp ~/P/r9 (mangopi)> ping d1
PING d1 (192.168.22.209) 56(84) bytes of data:
64 bytes from d1 (192.168.22.209): icmp_seq=1 ttl=64 time=0.626 ms
^C
--- dl ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.626/0.626/0.626/0.000 ms
dama@orangelemp ~/P/r9 (mangopi)> cpu-mqpro.sh /bbin/uname -a
Linux nezha 5.19.0-16561-g90c64258390f-dirty #306 SMP PREEMPT Sat Aug 27 15:37:13 CEST 2022 riscv64 (none)
dama@orangelemp ~/P/r9 (mangopi)> cpu-mqpro.sh /bbin/kexec ~/Projects/r9/target/riscv64-unknown-none-elf/debug/riscv64
[
]

1970/01/02 00:08:20
CPUD:orphan reaper: returns with 94
failed to put myself in foreground: inappropriate ioctl for device
#
/# ip a
1: lo: <UP,LOOPBACK> mtu 65536 state UNKNOWN
   link/loopback
   inet 127.0.0.1 scope host lo
       valid_lft forever preferred_lft forever
   inet6 ::1 scope host
       valid_lft forever preferred_lft forever
2: usb0: <UP,BROADCAST,MULTICAST> mtu 1500 state UP
   link/ether 02:3b:00:70:d1:0a
   inet 192.168.22.209 brd 192.168.22.255 scope global usb0
       valid_lft forever preferred_lft forever
   inet6 fe80::3b:ff:fe70:d10a scope link
       valid_lft forever preferred_lft forever
failed to put myself in foreground: inappropriate ioctl for device
#
/# [ 4.134514] u-root init: error creating mount -t "efivarfs" -y
[ 228.061891] resource: __walk_iomem_res_desc: from 0 to 184467440
[ 228.082155] kexec image: The entry point of kernel at 0x40200000
[ 228.089025] resource: __walk_iomem_res_desc: from 0 to 184467440
[ 228.099825] resource: __walk_iomem_res_desc: from 0 to 184467440
[ 228.107259] kexec image: Loaded device tree at 0x5fff6000
[ 228.853957] kexec_core: Starting new kernel
[ 228.858298] Will call new kernel at 5fffda30 from hart id 0
[ 228.863898] FDT image at 5fff6000
[ 228.867240] Bye...

r9 from the Internet
dtb at 0x5fff6000
hartid 0
panic in '/home/dama/.cargo/git/checkouts/rustsbi-53f9950f3fec986f/9
Some(internal error: entered unreachable code)
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.8 | VT102 | Offlin
"cpu-mqpro.sh /bbin/ke" 22:43 06-Sep-22

```



# R9 DEMO



Small computers everywhere



# Small computers everywhere

- 🌐 MCUs getting closer to application processors





# Small computers everywhere

- MCUs 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...



# 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

```
tick
RISC-V
vendor unknown
impl. ID: Mimpid { bits: 3697836033 }
ISA: Misa { bits: 1083183397 }

The meaning of life is to rewrite everything in Rust. 🦀🦀
Without love, breath is just a clock ticking. Type something!

tock[]
```

<https://github.com/orangecms/ch32v307-rust>



# MnemOS

---

<sup>9</sup><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”<sup>9</sup>*

---

<sup>9</sup><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”<sup>9</sup>*

```
oreboot 🍌
DDR3@792MHz
test OK
512M 🍌
NOR flash: c2/2018
Load... 🍌
---
Run payload at 0x40000000
Bootstrapping Kernel...
Heap Start: 0000000048000000
Heap Size: 0000000018000000
Kernel configured. Waiting for initialization...
Initalized. Starting Run Loop.
[TASK 0, ct 00000] lol. lmao.
[TASK 0, ct 00001] lol. lmao.
[TASK 0, ct 00002] lol. lmao.
[TASK 1, ct 00000] beep, boop.
[TASK 0, ct 00003] lol. lmao.
[TASK 0, ct 00004] lol. lmao.
[TASK 0, ct 00005] lol. lmao.
[TASK 1, ct 00001] beep, boop.
```

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

---

<sup>9</sup><https://onevariable.com/blog/mnemos-moment-1/>



# Gadget hacking and development boards





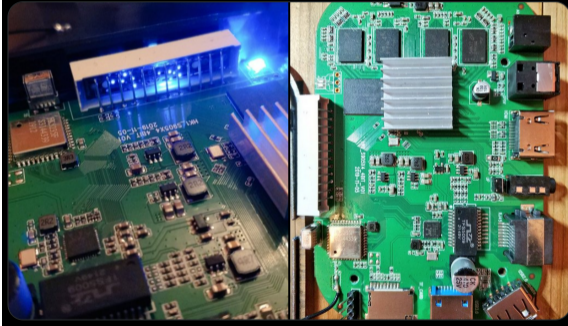
# Gadget hacking and development boards



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. :-)



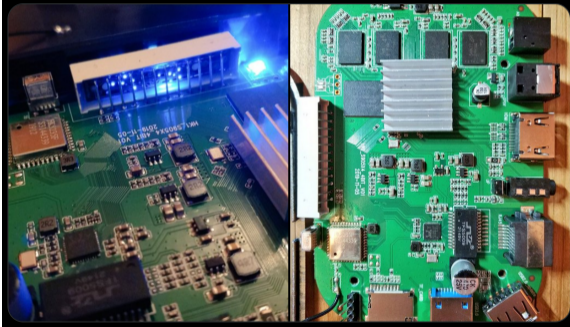
# Gadget hacking and development boards



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. :-)



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



# User Interface Design

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

---

<sup>10</sup>[http://doc.cat-v.org/plan\\_9/1st\\_edition/help/help.pdf](http://doc.cat-v.org/plan_9/1st_edition/help/help.pdf)



# User Interface Design

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

A Minimalist Global User Interface<sup>10</sup> (1991)

combo of editor + window system + shell + user interface

---

<sup>10</sup>[http://doc.cat-v.org/plan\\_9/1st\\_edition/help/help.pdf](http://doc.cat-v.org/plan_9/1st_edition/help/help.pdf)

<sup>11</sup>[https:](https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces)

[//www.researchgate.net/publication/2487366\\_Genuinely\\_Functional\\_User\\_Interfaces](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<sup>10</sup> (1991)

combo of editor + window system + shell + user interface

Genuinely Functional User Interfaces<sup>11</sup> (2001)

GUI library for Haskell based on formal model of user interfaces

---

<sup>10</sup>[http://doc.cat-v.org/plan\\_9/1st\\_edition/help/help.pdf](http://doc.cat-v.org/plan_9/1st_edition/help/help.pdf)

<sup>11</sup>[https:](https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces)

[//www.researchgate.net/publication/2487366\\_Genuinely\\_Functional\\_User\\_Interfaces](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<sup>10</sup> (1991)

combo of editor + window system + shell + user interface

Genuinely Functional User Interfaces<sup>11</sup> (2001)

GUI library for Haskell based on formal model of user interfaces

Touch input

Contemporary devices commonly feature touch screens.

---

<sup>10</sup>[http://doc.cat-v.org/plan\\_9/1st\\_edition/help/help.pdf](http://doc.cat-v.org/plan_9/1st_edition/help/help.pdf)

<sup>11</sup>[https://www.researchgate.net/publication/2487366\\_Genuinely\\_Functional\\_User\\_Interfaces](https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces)

<sup>12</sup><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<sup>10</sup> (1991)

combo of editor + window system + shell + user interface

Genuinely Functional User Interfaces<sup>11</sup> (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.<sup>12</sup>

---

<sup>10</sup>[http://doc.cat-v.org/plan\\_9/1st\\_edition/help/help.pdf](http://doc.cat-v.org/plan_9/1st_edition/help/help.pdf)

<sup>11</sup>[https://www.researchgate.net/publication/2487366\\_Genuinely\\_Functional\\_User\\_Interfaces](https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces)

<sup>12</sup><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<sup>10</sup> (1991)

combo of editor + window system + shell + user interface

Genuinely Functional User Interfaces<sup>11</sup> (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.<sup>12</sup>

How about a three-finger diagonal to draw an app launcher?

---

<sup>10</sup>[http://doc.cat-v.org/plan\\_9/1st\\_edition/help/help.pdf](http://doc.cat-v.org/plan_9/1st_edition/help/help.pdf)

<sup>11</sup>[https:](https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces)

[//www.researchgate.net/publication/2487366\\_Genuinely\\_Functional\\_User\\_Interfaces](https://www.researchgate.net/publication/2487366_Genuinely_Functional_User_Interfaces)

<sup>12</sup><https://sailfishos.org/design/gestures/>



Thank you! :)



## Related

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

<https://metaspora.org/harvey-os-lni2020.pdf>

Drivers from Outer Space (CLT 2022)

<https://chemnitzer.linux-tage.de/2022/en/programm/beitrag/226>

IDROS - Instrumented Distributed Resources OS

<https://metaspora.org/idros.pdf>

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

<https://media.ccc.de/v/fire-shonks-2022-49154-platform-system-interface-design-und-evaluation-holistischer-computerarchitektur>



# Follow Me



Daniel Maslowski

<https://github.com/orangecms>

<https://twitter.com/orangecms>

<https://mastodon.social/@cyrevolt>

<https://youtube.com/@cyrevolt>

<https://twitch.tv/cyrevolt>

<https://metaspora.org/clueless-os-dev-iger2023.pdf>

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

