

HelenOS in the Year of the Pig

<http://www.helenos.org>

Jakub Jermář
jakub@jermar.eu



HelenOS

HelenOS in a nutshell



- Portable microkernel-based multiserer OS
- Designed and implemented from scratch
- Not a clone of any existing OS / API
- Virtually no third-party code
- Fine-grained userspace components

Since last FOSDEM...

May

HelenOS 0.9.1 Armonia

L4Re::Treat<2019>

Jun / July

SUSE Hack Week 19: HelenOS of the <future>

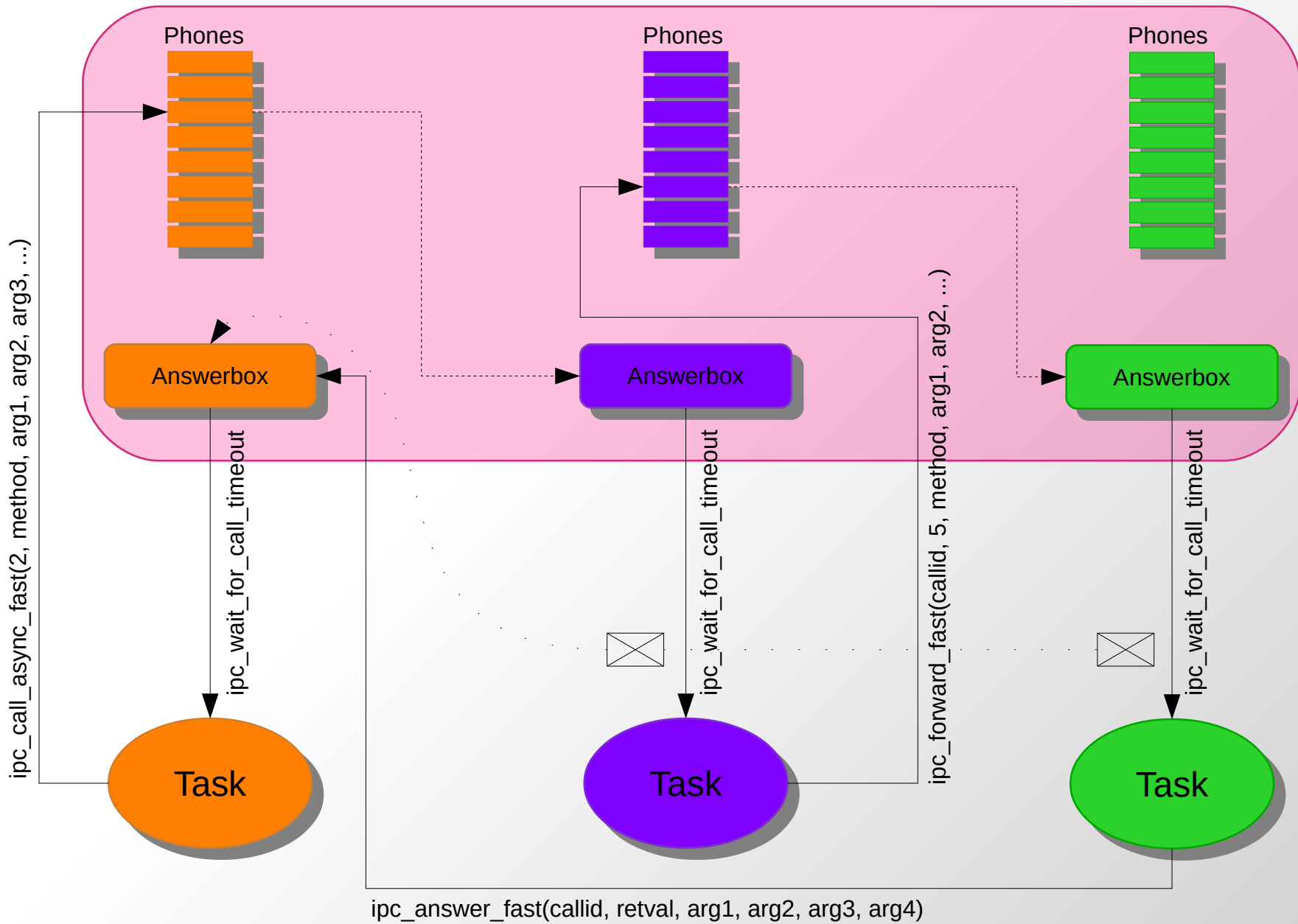
August

HelenOS quasi-Camp 2019

Master and beyond

- C++ <future> header (master)
- Switch to the Meson build system (master)
- IPC v2 (WiP, branch)
- HelenOS as anykernel (WiP, branch)
- Display server (WiP, branch)

HelenOS IPC Recap



HelenOS IPC Recap



Call

non-reusable, fixed-size, kernel allocated/managed

Phone

(unnecessary?) level of indirection between client and server

Answerbox

asynchronous, waitable

(still) one-per task

Where should the answer go with >1 answerboxes?

How does a server wait for multiple requests?

How does a client wait for multiple answers?

IPC buffer

preallocated, reusable, arbitrary size, waitable

IPC endpoint

asynchronous, waitable

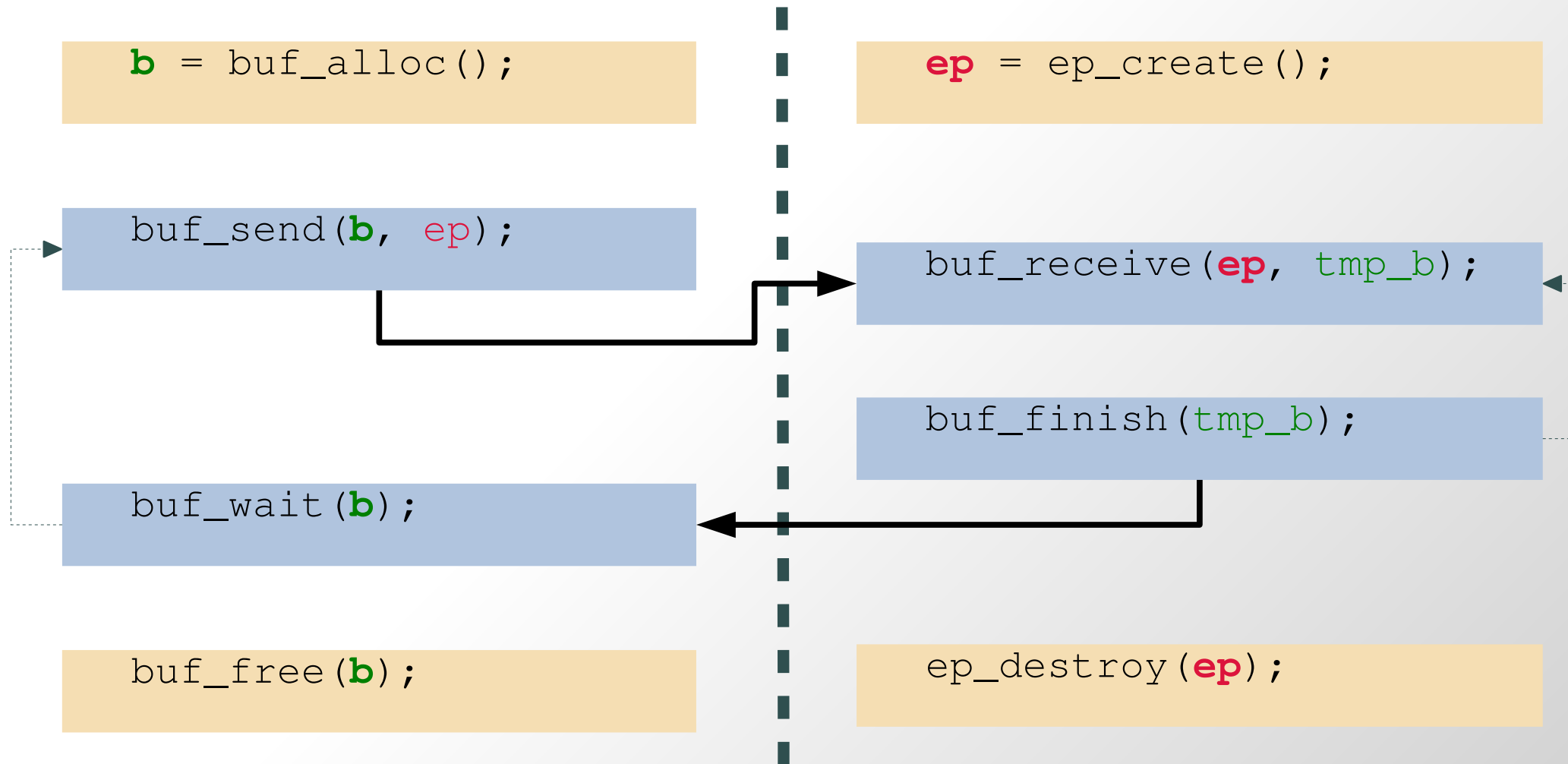
Caplists

container for other kernel objects

similar to Mach port sets / epoll

waiting for multiple requests / answers

Simple Roundtrip



Multiple buffers and endpoints

```
bcl = caplist_create();  
b1 = buf_alloc();  
b2 = buf_alloc();
```

```
buf_send(b1, ep1, bcl);  
buf_send(b2, ep2, bcl);
```

```
buf_wait(bcl);  
buf_wait(bcl);
```

```
buf_free(b1);  
buf_free(b2);  
caplist_destroy(bcl);
```

```
ec1 = caplist_create();  
ep1 = ep_create(ec1);  
ep2 = ep_create(ec1);
```

```
buf_receive(ec1, tb);
```

```
buf_finish(tb);
```

```
ep_destroy(ep1);  
ep_destroy(ep2);  
caplist_destroy(ec1);
```

HelenOS as Anykernel (WiP)



Martin Děcký @mdecky · Aug 17, 2019

The first prototype of HelenOS **anykernel** runs on my machine. Compared to the NetBSD rumpkernel approach, our foundation is the fine-grained microkernel multiserver architecture. The goal is to have the maximum flexibility of the OS architecture at deployment time.

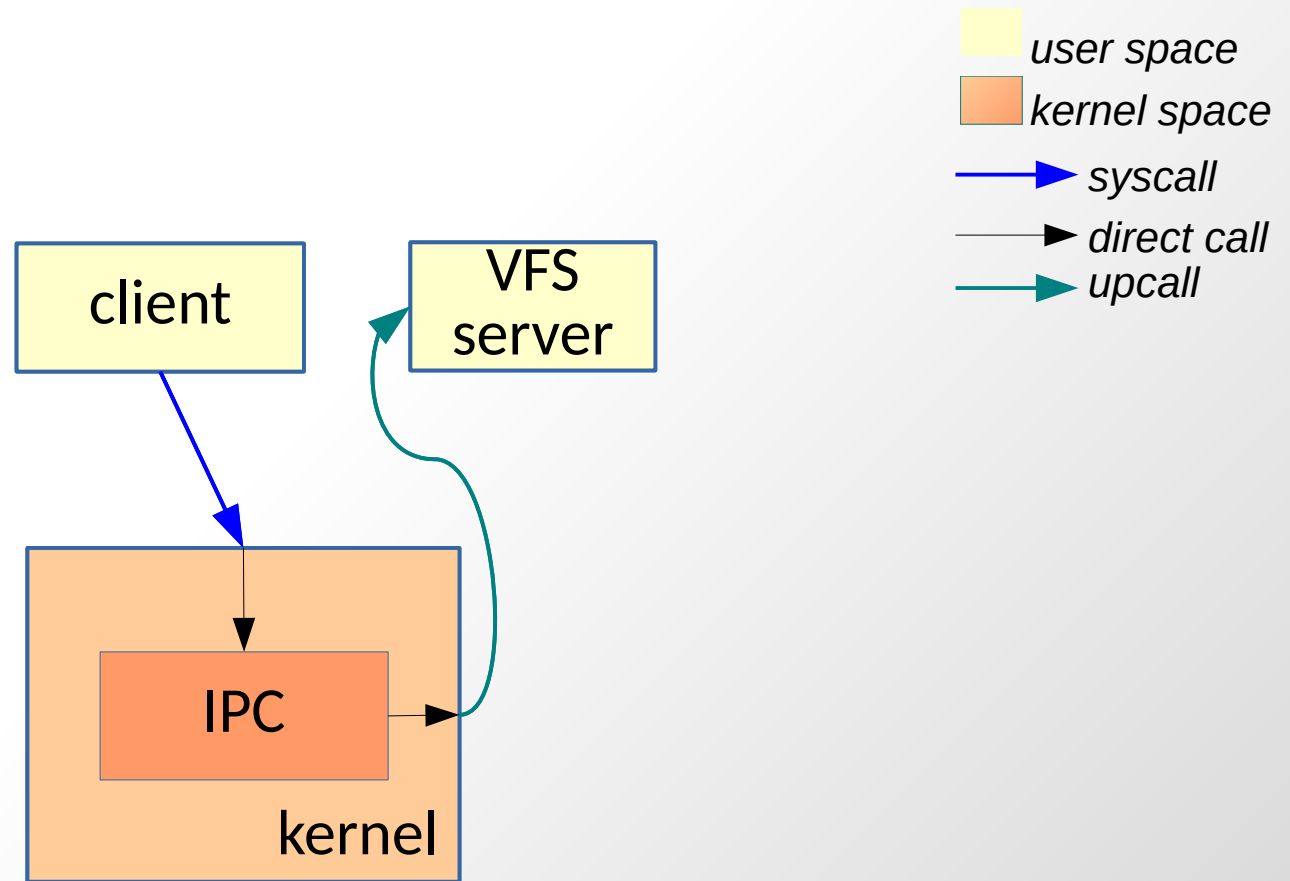
2

4

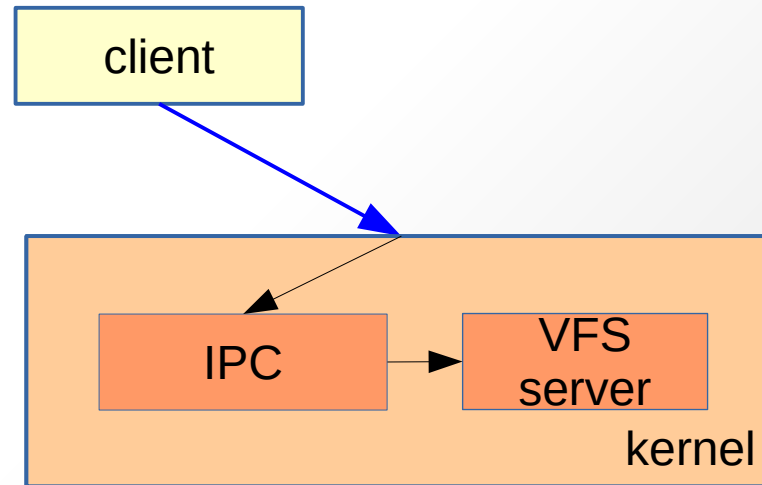
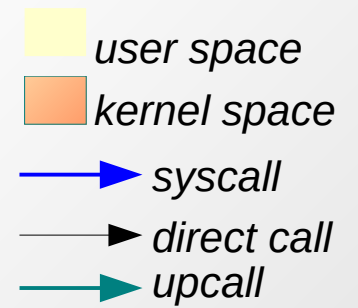
15



HelenOS as Anykernel



HelenOS as Anykernel



Display server (WiP)

- Eventually to replace the current compositor
- More modular structure than compositor
- Compositor needs memory for all windows
- Better control of rendering and buffering policies
- HelenOS IPC is just a transport
- Different pixel formats, text mode
- And more, see ticket #816

- <http://www.helenos.org/wiki/ReleaseNotes/0.9.1>
- https://blog.kernkonzept.com/retreat/social/2019/06/06/14re_treat.html
- <https://hackweek.suse.com/19/projects/helenos-of-the>
- <https://github.com/jxsvoboda/helenos/tree/gfx>
- <http://www.helenos.org/ticket/816>
- <https://github.com/jermar/helenos/tree/ipc>

<http://www.helenos.org>

<https://github.com/HelenOS/helenos>

@HelenOSOrg

@jjermar

Thank you!