

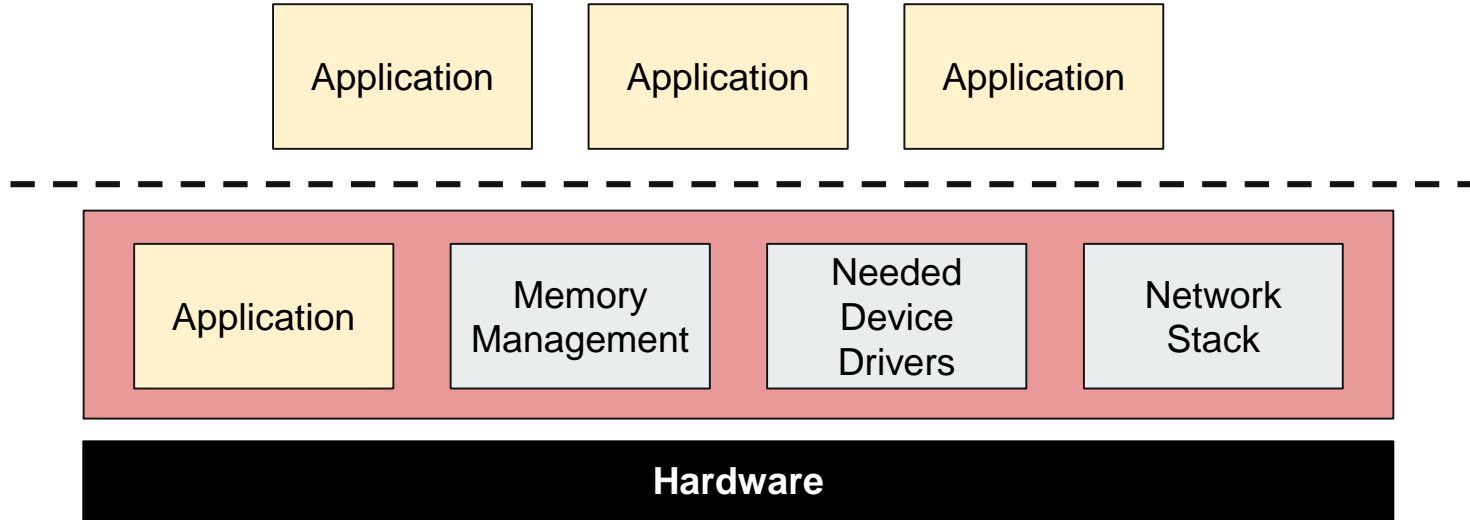


UKL: A Unikernel based on Linux

Ali Raza, James Cadden, Tommy Unger, Parul Sohal,
Ulrich Drepper, Richard Jones, Orran Krieger,
Larry Woodman



Nonlinear Kernels



EbbRT

More than 2X improved
Memcached throughput from Linux

Advantages of Unikernels

- Light weight
 - Small attack surface
 - Faster boot times
- Improved performance
 - Avoid ring transitions
 - Application-specific optimizations

The logo for LING, consisting of the word "LING" in white capital letters on a green rectangular background.

LING

Its website takes
25 MB memory

The logo for ClickOS, featuring the word "ClickOS" in white capital letters on a teal rectangular background.

ClickOS

Produces network software that can
process 5 million packets/s, boot
time under 30 ms

The logo for includeOS, featuring a blue hexagonal icon with a white cube inside, followed by the text "includeOS" in white on a dark blue background.

includeOS

Base VM starts at 1 MB size



Rumprun

The logo for MIRAGE OS, featuring the word "MIRAGE" in white and "OS" in green on a black rectangular background.

MIRAGE OS

DNS compiles into 449 KB



Problem: Development Model

- Clean slate
- Fork of an existing code base

Issues

- Maintenance/testing nightmare
- Lack of community

BU & Red Hat collaborative



New Development Model

Advantages

- Part of Linux and GlibC
- Incremental optimizations
- Maintained by community
- Re-use of the entire Linux and glibc code base!
- Unchanged Linux API guarantee to developers
- Support of all the existing devices drivers, file systems etc.
- VMs or bare metal deployment

Imagine a unikernel with GPU support!



Is it even possible?

Summer 2018



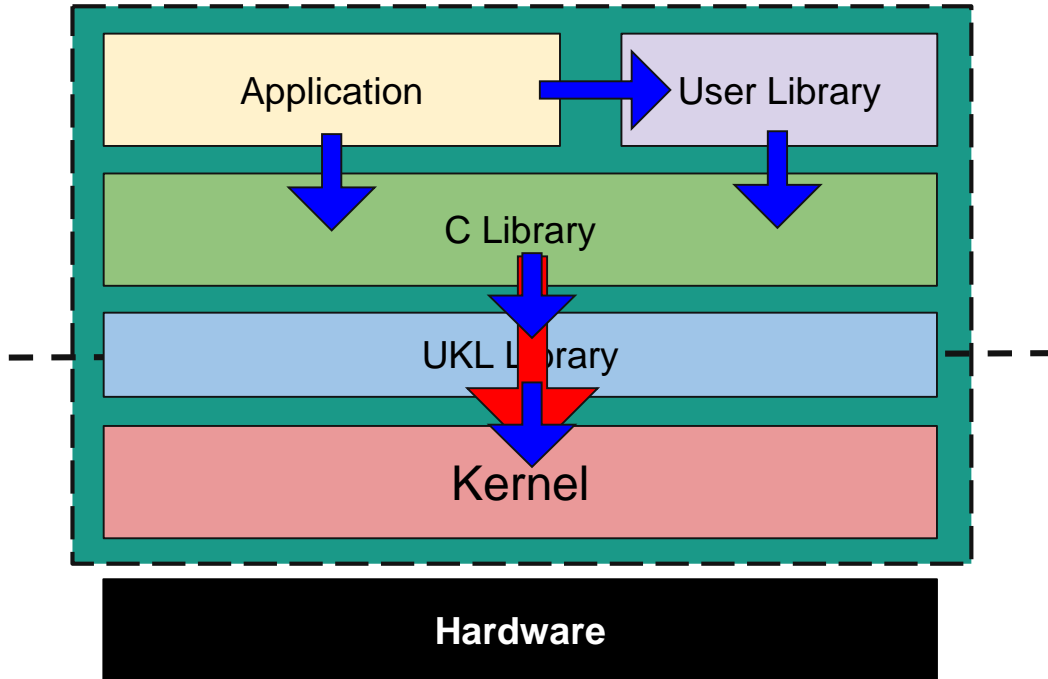


Implementation Details

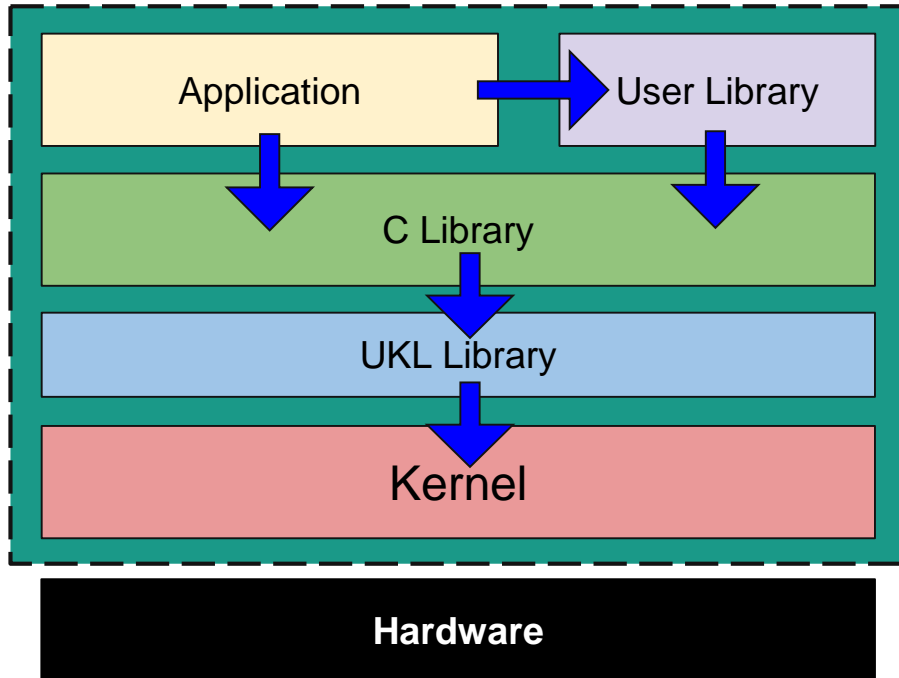
- Minimal changes to Linux code
- Modest changes in glibc in a separate subtree

```
/*  
 * linux/init/main.c  
 */  
...  
  
static int run_init_process(const char *init_filename)  
{  
    kmain();  
  
    /*argv_init[0] = init_filename;  
    return do_execve(getname_kernel(init_filename),  
                    (const char __user *const __user *)argv_init,  
                    (const char __user *const __user *)envp_init);*/  
}  
  
...
```

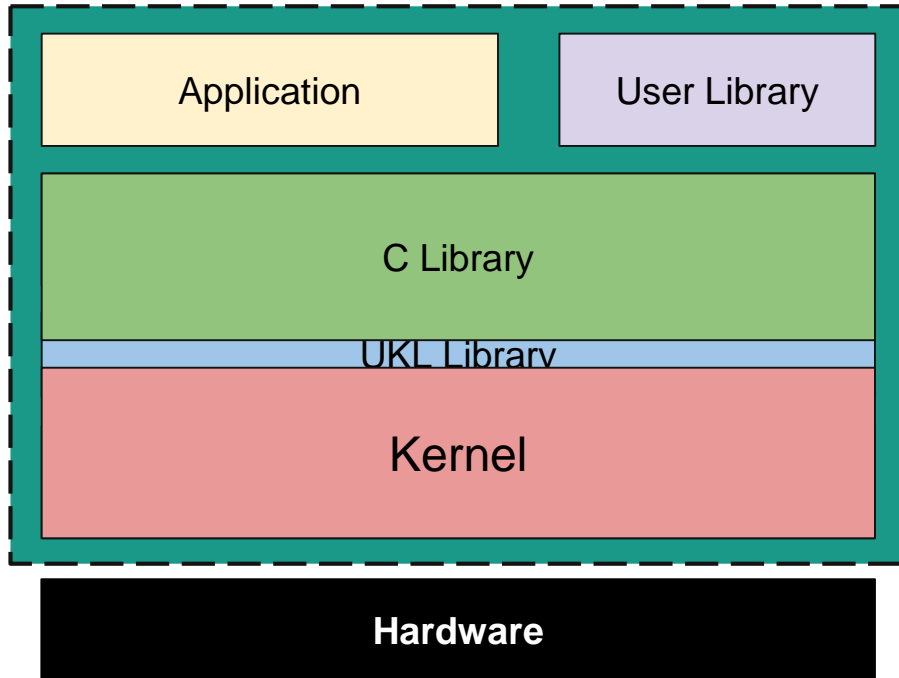

UKLinux Architecture



UKL Architecture



UKL Architecture





Next Steps

- Pthreads
 - Threading model and thread local storage
- Memcached
- Upstream acceptance



Research Questions

- Performance and start time advantages
- Add optimizations from specialized unikernels
- Explore security advantages

Use cases

- Cloud workloads e.g., Memcached
- Functions as a service
- Explore different use cases, e.g., HPC, embedded
- Optimize I/O intensive applications e.g., Ceph which uses SPDK and DPDK