



Ninjastorms: Team Kernel

Felix Roth und Konrad Hanff

<https://github.com/hpi-bs2-st2020-ninjastorms-kernel/ninjastorms>

ursprüngliche Ziele

- Hardwareabstraktion
- Interrupts
- Prozesstruktur
- Syscalls
- Koordination mit anderen Gruppen

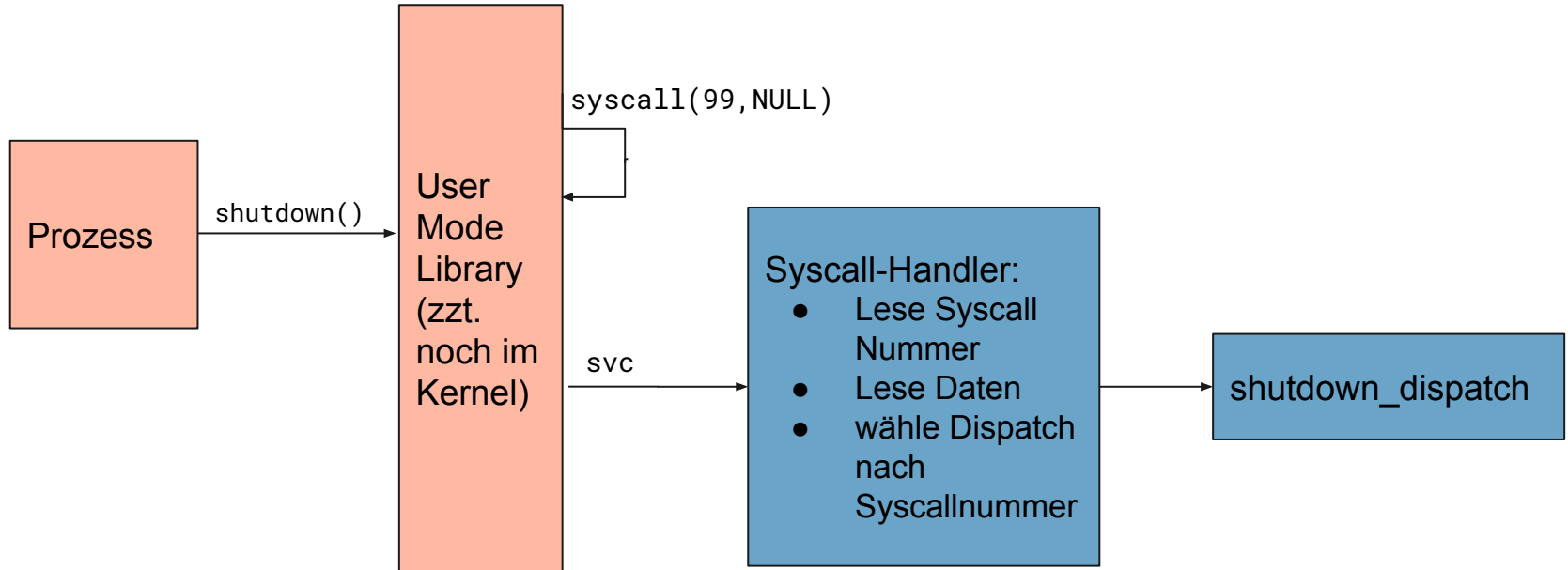


Bildquelle: Lego

Fortschritt



Syscalls und Interrupts





How-To: neuen Syscall einfügen

- Kernel Mode:
 - Syscall Nummer und symbolischen Namen in `syscall_handler.h` einfügen
 - Funktion in `syscall_handler.c`: `syscall_dispatcher(unsigned int syscallno, void *data)` mit richtiger Syscall Nummer hinzufügen
- User Mode: API Funktion ruft:
 - `syscall(unsigned int number, void* data)` mit beliebiger Struktur in data
 - auf data kann im Kernel Mode zugegriffen werden



Prozesse/Tasks

- Prozesse arrangiert in Array im Kernel
- Jeder Prozess hat
 - Ausführungsumgebung (Register, Programcounter, ...)
 - `pid`
 - `parent_pid`
 - `IPC_Buffer`
 - `stored_errno` -> Prozessspezifische errno
- Prozesse können durch Syscalls miteinander interagieren



Interprozesskommunikation: Queue

- Jeder Prozess hat ein Buffer im Kernel
- Prozess kann Buffer öffnen
- Andere Prozesse können an spezielle pid senden
- Empfänger kann aus Buffer lesen und konsumieren



Unsere API

- `create_process(void * function)`
- `exit()`
- `get_pid()`
- `get_parent_pid()`
- `kill(pid_t target)`
- `is_predecessor(pid_t child, pid_t pred)`
- `ipc_buffer_open()`
- `ipc_buffer_close()`
- `ipc_buffer_send(int value, pid_t target)`
- `ipc_buffer_read()`
- `ipc_buffer_length()`
- `print_tasks_info()`
- `shutdown()`

Weitere Ziele

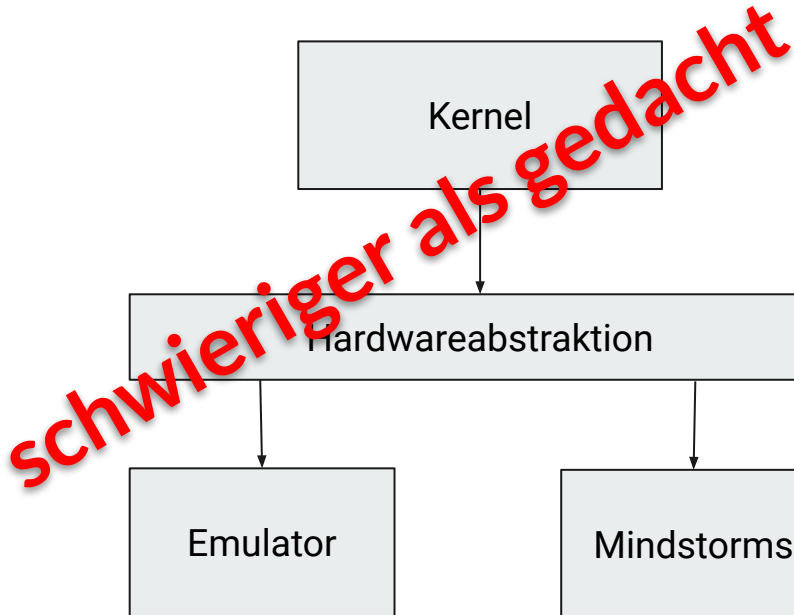




Schwierigkeiten

- Mit `exit()` nicht den Speicher zuschreiben
- Tatsächliches `shutdown()`
- Debugging

Hardwareabstraktion



Interface für Timer, Speicher
(Memory-mapped registers),
Geräte, ...

tatsächliche
Speicheradressen



Weitere Herausforderungen und Schritte

- C Standardbibliothek weiter implementieren
- Bessere Interprozesskommunikation
- Aufräumen, Refactoring, Dokumentation
- **Nutzerinteraktion am Terminal**
 - **Filedeskriptoren**
 - **Treiber für Tastatureingabe**
- Integration mit Memorygruppe