

The pin-init API Training

Benno Lossin (benno.lossin@proton.me)

Kangrejós, 16.09.2023

Outline

- 1 Initializing Data In-Place
 - Mutexes in Rust
 - Why does Mutex need `pin-init`?
 - A more Complex Example
- 2 What about Fallibility?
- 3 Zeroing Data
- 4 Drop Support
- 5 The Unsafe API

Before We Start

Before We Start

- this is a training, so questions and discussion are welcomed and wanted!

Before We Start

- this is a training, so questions and discussion are welcomed and wanted!
- to download the training files do:

```
git remote add benno https://github.com/y86-dev/linux.git
git fetch benno kangrejos-2023-training
git checkout kangrejos-2023-training
```

Before We Start

- this is a training, so questions and discussion are welcomed and wanted!
- to download the training files do:

```
git remote add benno https://github.com/y86-dev/linux.git
git fetch benno kangrejos-2023-training
git checkout kangrejos-2023-training
```
- no worries, I will start with some slides and the training files already contain all of that.

Mutexes in Rust

Mutexes in Rust

C

```
struct counter {  
    struct mutex lock;  
    int value;  
};
```

Rust

Mutexes in Rust

C

```
struct counter {  
    struct mutex lock;  
    int value;  
};
```

Rust

```
type Counter = Mutex<i32>;
```

Mutexes in Rust

C

```
struct counter {  
    struct mutex lock;  
    int value;  
};
```

```
struct counter *ctr = kmalloc(...);  
if (!ctr)  
    return -ENOMEM;  
mutex_init(&ctr->lock);  
ctr->value = 42;
```

Rust

```
type Counter = Mutex<i32>;
```

Mutexes in Rust

C

```
struct counter {  
    struct mutex lock;  
    int value;  
};
```

```
struct counter *ctr = kmalloc(...);  
if (!ctr)  
    return -ENOMEM;  
mutex_init(&ctr->lock);  
ctr->value = 42;
```

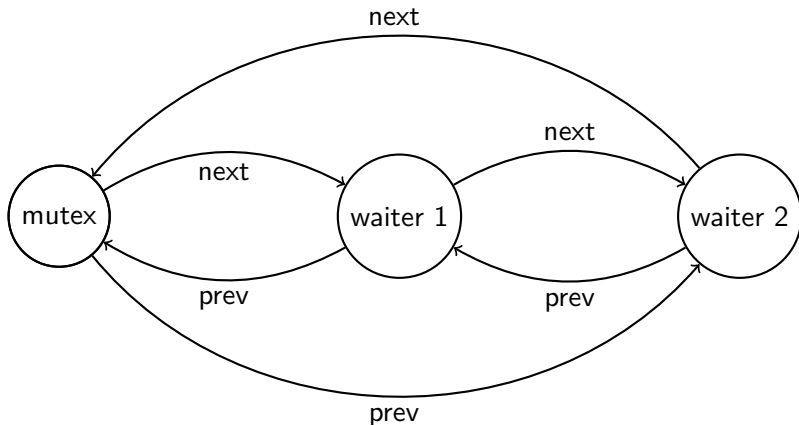
Rust

```
type Counter = Mutex<i32>;
```

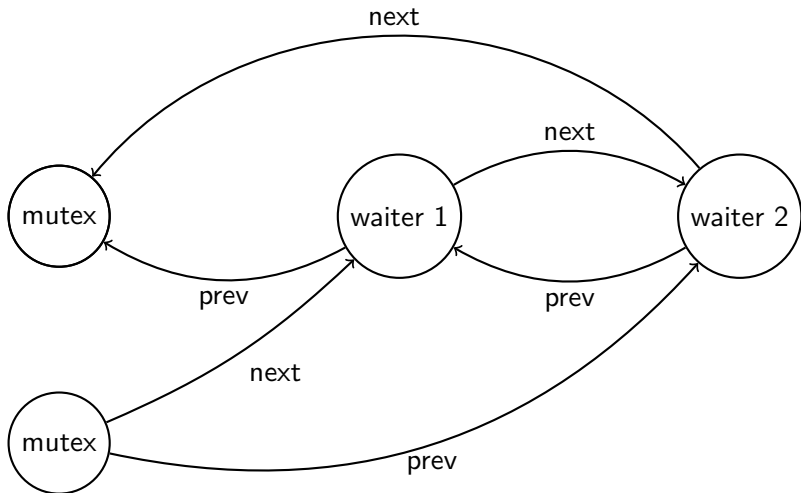
```
let counter =  
    Box::pin_init(new_mutex!(42))?;
```

Why does Mutex need pin-init?

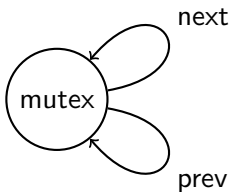
Why does Mutex need pin-init?



Why does Mutex need pin-init?



Why does Mutex need pin-init?



A more Complex Example

```
struct named_counter {  
    char *name;  
    struct mutex lock;  
    int cur_value;  
    int max_value;  
};
```


A more Complex Example

```
struct named_counter {  
    char *name;  
    struct mutex lock;  
    int cur_value;  
    int max_value;  
};
```

```
struct NamedCounter {  
    name: &'static str,  
    value: Mutex<(i32, i32)>,  
}
```

A more Complex Example

```
struct named_counter {
    char *name;
    struct mutex lock;
    int cur_value;
    int max_value;
};

void named_counter_init(
    struct named_counter *ctr,
    char *name,
    int cur_value,
    int max_value
) {
    ctr->name = name;
    ctr->cur_value = cur_value;
    ctr->max_value = max_value;
    mutex_init(&ctr->lock);
}
```

```
struct NamedCounter {
    name: &'static str,
    value: Mutex<(i32, i32)>,
}
```

A more Complex Example

```
struct named_counter {
    char *name;
    struct mutex lock;
    int cur_value;
    int max_value;
};

void named_counter_init(
    struct named_counter *ctr,
    char *name,
    int cur_value,
    int max_value
) {
    ctr->name = name;
    ctr->cur_value = cur_value;
    ctr->max_value = max_value;
    mutex_init(&ctr->lock);
}
```

```
struct NamedCounter {
    name: &'static str,
    value: Mutex<(i32, i32)>,
}

fn new(
    name: &'static str,
    cur_value: i32,
    max_value: i32,
) -> impl PinInit<Self> {
```

A more Complex Example

```
struct named_counter {
    char *name;
    struct mutex lock;
    int cur_value;
    int max_value;
};

void named_counter_init(
    struct named_counter *ctr,
    char *name,
    int cur_value,
    int max_value
) {
    ctr->name = name;
    ctr->cur_value = cur_value;
    ctr->max_value = max_value;
    mutex_init(&ctr->lock);
}
```

```
struct NamedCounter {
    name: &'static str,
    value: Mutex<(i32, i32)>,
}

fn new(
    name: &'static str,
    cur_value: i32,
    max_value: i32,
) -> impl PinInit<Self> {
    pin_init!(Self {
        name,
        value <- new_mutex!(
            (cur_value, max_value)
        ),
    })
}
```

A more Complex Example

```

struct named_counter {
    char *name;
    struct mutex lock;
    int cur_value;
    int max_value;
};

void named_counter_init(
    struct named_counter *ctr,
    char *name,
    int cur_value,
    int max_value
) {
    ctr->name = name;
    ctr->cur_value = cur_value;
    ctr->max_value = max_value;
    mutex_init(&ctr->lock);
}

```

```

#[pin_data]
struct NamedCounter {
    name: &'static str,
    #[pin]
    value: Mutex<(i32, i32)>,
}

fn new(
    name: &'static str,
    cur_value: i32,
    max_value: i32,
) -> impl PinInit<Self> {
    pin_init!(Self {
        name,
        value <- new_mutex!(
            (cur_value, max_value)
        ),
    })
}

```

Initializing NamedCounter

```
let ctr_init = NamedCounter::new(  
    "My Counter",  
    0,  
    42,  
);
```

Initializing NamedCounter

```
let ctr_init = NamedCounter::new(  
    "My Counter",  
    0,  
    42,  
);
```

- on the heap, uniquely owned:

```
let ctr = Box::pin_init(ctr_init);
```

Initializing NamedCounter

```
let ctr_init = NamedCounter::new(  
    "My Counter",  
    0,  
    42,  
);
```

- on the heap, uniquely owned:

```
let ctr = Box::pin_init(ctr_init);
```
- on the heap, sharable between threads:

```
let ctr = Arc::pin_init(ctr_init);
```


Initializing NamedCounter

```
let ctr_init = NamedCounter::new(  
    "My Counter",  
    0,  
    42,  
);
```

- on the heap, uniquely owned:

```
let ctr = Box::pin_init(ctr_init);
```
- on the heap, sharable between threads:

```
let ctr = Arc::pin_init(ctr_init);
```
- on the stack:

```
stack_pin_init!(let ctr = ctr_init);
```

What about Fallibility?

Code Demo

```
git remote add benno https://github.com/y86-dev/linux.git  
git fetch benno kangrejos-2023-training  
git checkout kangrejos-2023-training
```

Zeroing Data

Code Demo

```
git remote add benno https://github.com/y86-dev/linux.git  
git fetch benno kangrejos-2023-training  
git checkout kangrejos-2023-training
```

Drop Support

Code Demo

```
git remote add benno https://github.com/y86-dev/linux.git  
git fetch benno kangrejos-2023-training  
git checkout kangrejos-2023-training
```

The Unsafe API

Code Demo

```
git remote add benno https://github.com/y86-dev/linux.git  
git fetch benno kangrejos-2023-training  
git checkout kangrejos-2023-training
```