CHERITech'25

CONFERENCE

Manchester, UK

2025

14th November

cheri-alliance.org

CHERI

Redesigning Thread-Local Storage For CHERI

Jessica Clarke
University of Cambridge

Disclaimers

- There will be:
  - Assembly
  - Pointers
  - Diagrams showing multiple levels of pointers
- I only have 15 minutes; many details omitted
- Find me afterwards if you still want to know more…

Background

CHERI

What Is TLS?

- GNU + Microsoft extensions
- Standardised in C11 / C++11

```
_Thread_local int x;

int next(void) {
    return ++x;
}
```

Thread 1 next → 1

Thread 1 next → 2

Thread 1 next → 3

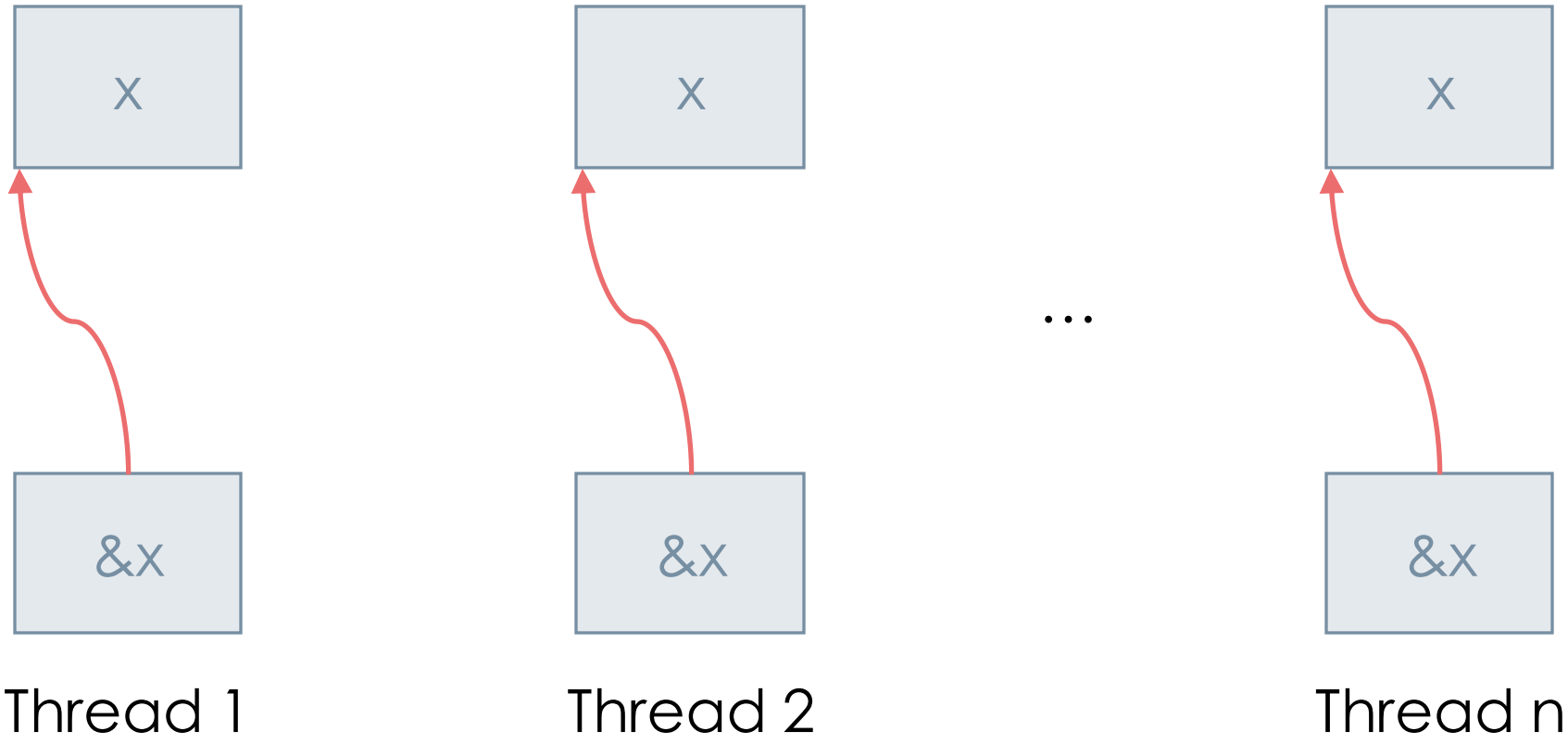
Thread 2 next → 1

Thread 2 next → 2

Thread 1 next → 4

…

What Is TLS?



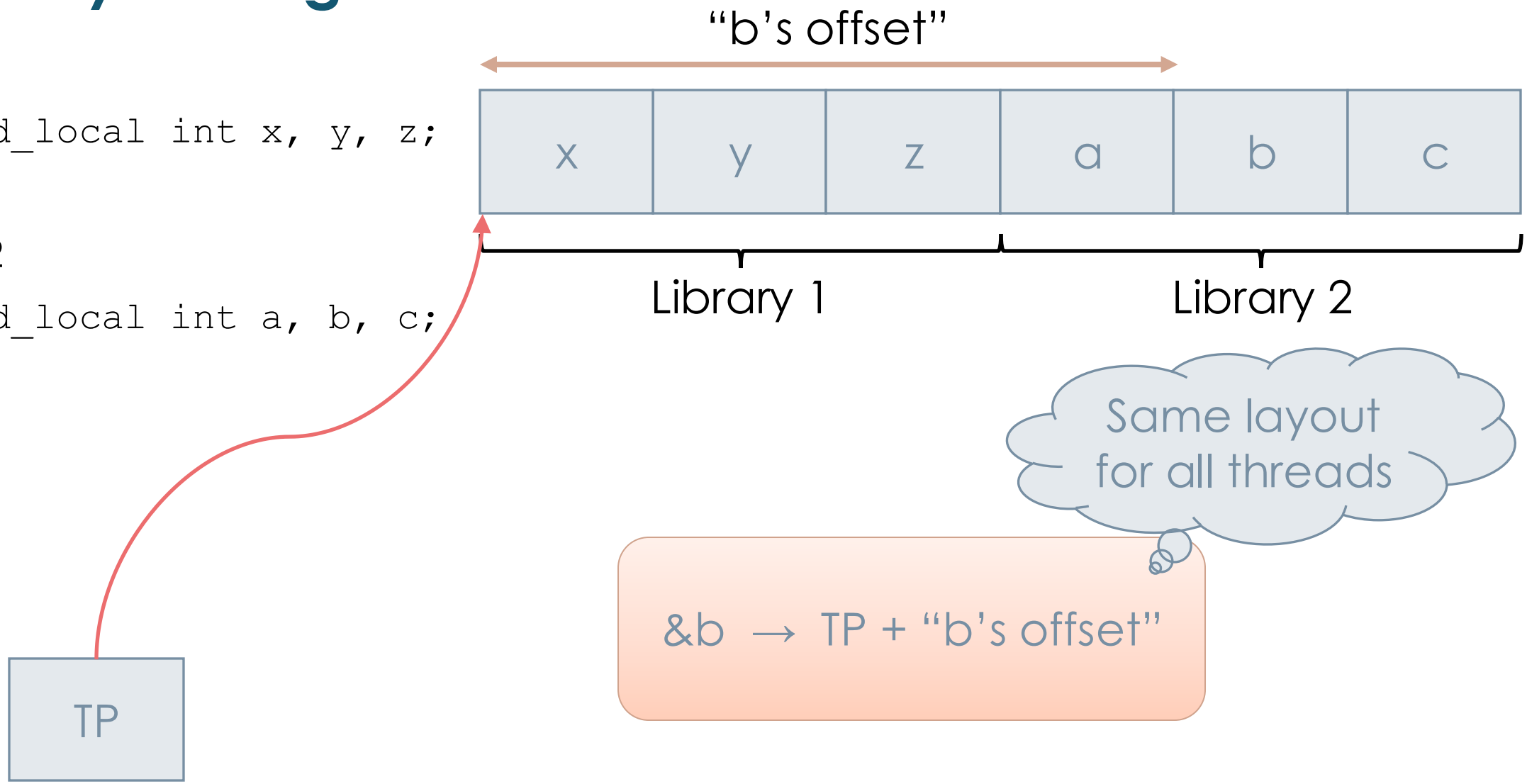
Today's Design

Library 1

```
_Thread_local int x, y, z;
```

Library 2

```
_Thread_local int a, b, c;
```



Where Do These Offsets Come From?

- Run-time loader first loads executable, then recursively all library dependencies
- Executable is always first
- Offsets for executable's TLS variables are known at link time – hard-coded into output binary

```
next:
    mrs  x8, TPIDR_EL0
    add  x8, x8, #0x0, lsl #12
    add  x8, x8, #0x10
    ldr  w9, [x8]
    add  w0, w9, #0x1
    str  w0, [x8]
    ret
```

TP

0x0 << 12 + 0x10

Load, increment, store

Where Do These Offsets Come From?

- In general, offset not known at link time
- Offset itself is a variable
- Stored in Global Offset Table ("GOT") alongside pointers to globals
- Run-time loader fills in

```
next:
    adrp x8, 0x20000
    ldr  x8, [x8, #0x8a8]
    mrs  x9, TPIDR_EL0
    ldr  w10, [x9, x8]
    add  w0, w10, #0x1
    str  w0, [x9, x8]
    ret
```

Load offset

Variable offset

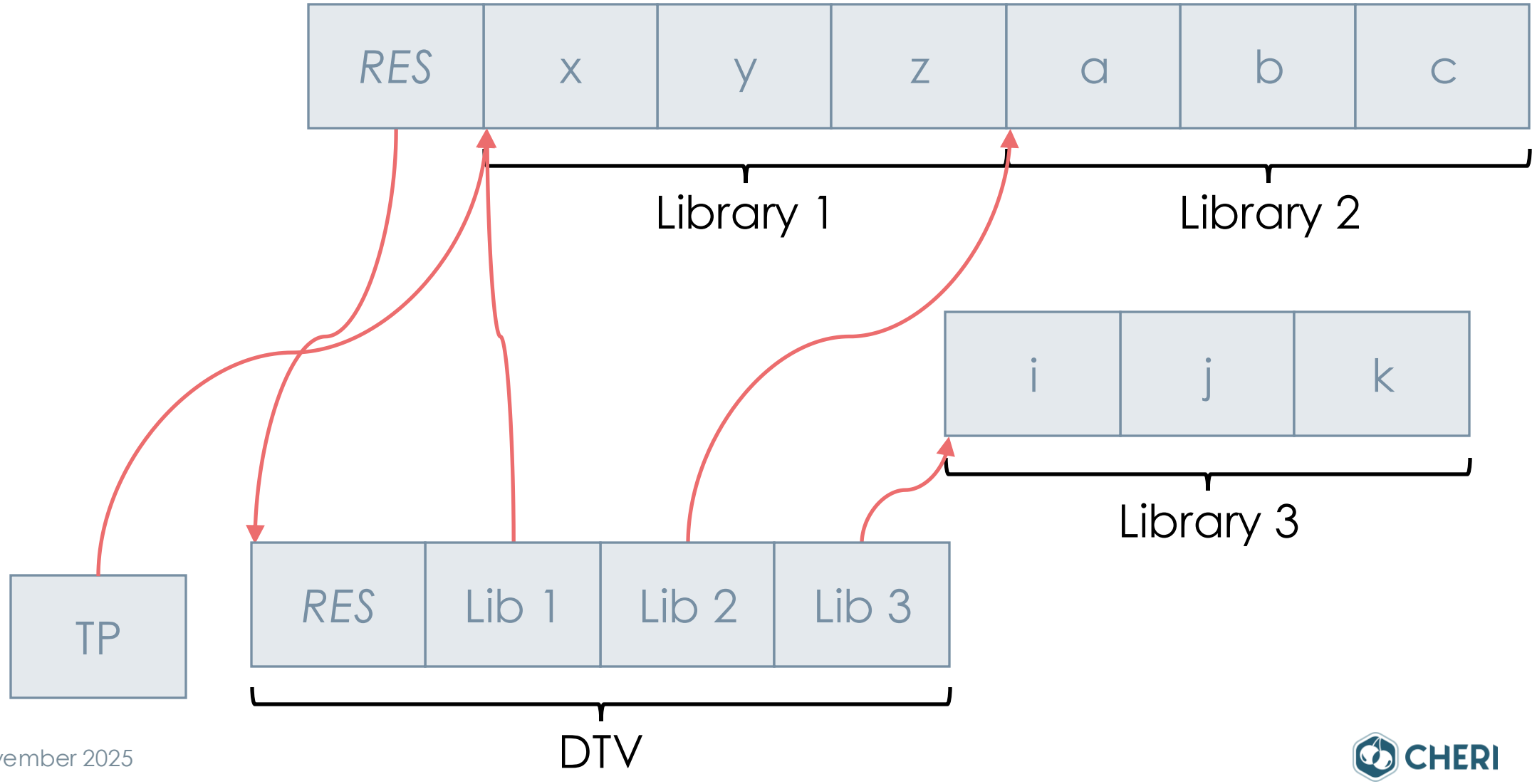
dlopen()

- Dynamically loads new libraries after program has started running (e.g. plugins)
- What if they define new thread-local variables?
  - Space already allocated for existing threads based on libraries initially present
  - Growing allocation might require moving it; pointers to thread-local variables would no longer work

dlopen()

- New variables stored in separate allocation
- How to get to them? Cannot just add offset to TP any more
- Call magic “__tls_get_addr()” function implemented by run-time loader
- Run-time loader reserves space at start of TLS block to track these allocations (points to “Dynamic Thread Vector”)

dlopen()


RES
x
y
z
a
b
c
Library 1
Library 2
i
j
k
Library 3
TP
RES
Lib 1
Lib 2
Lib 3
DTV

TLS Meets CHERI

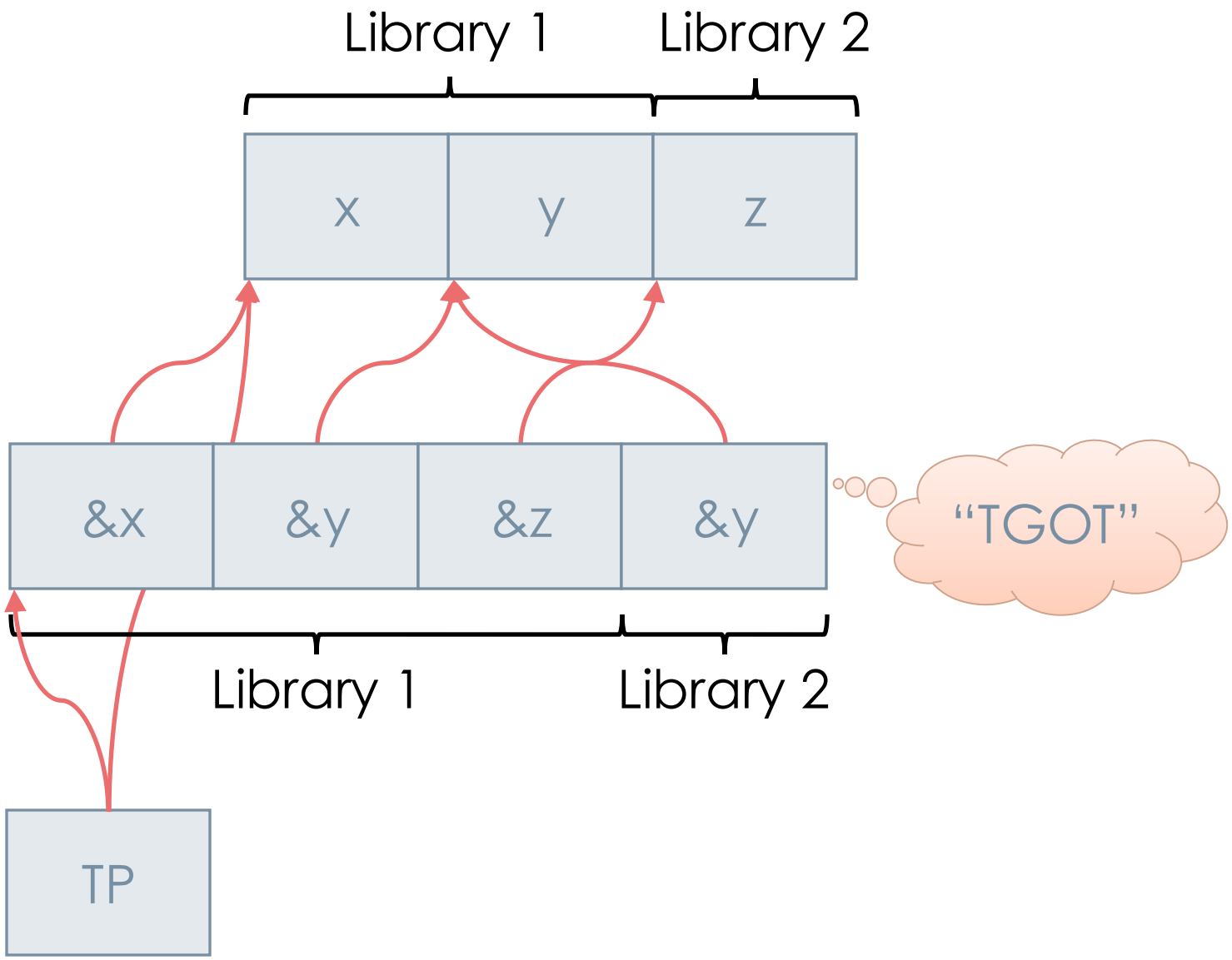
14 November 2025

CHERI

Initial (Current) CHERI Adaptation

- Same model
- Pointers now capabilities (TP and DTV entries, plus run-time loader's pointer to the DTV)
- Problems:
  - TP's bounds cover the entire TLS block – powerful capability
  - TP + offset for &x isn't bounded to just x – need to dynamically restrict bounds every time x is referenced
  - Ditto for DTV entries (though can at least restrict to library's subset)
  - Compartmentalisation?
  - 😟

Indirection



As Code

Constant Offset

```
next:
   mrs   c0, CTPIDR_EL0
   add   c0, c0, #0x0, lsl #12
   ldr   c1, [c0, #0x20]

   ldr   w8, [c1]
   add   w0, w8, #0x1
   str   w0, [c1]
   ret
```

0x0 << 12 + 0x20

Load, increment, store

Variable Offset

```
next:
   adrp c8, 0x20000
   ldr  x8, [c8, #0x550]
   mrs  c0, CTPIDR_EL0
   ldr  c1, [c0, x8]
   ldr  w8, [c1]
   add  w0, w8, #0x1
   str  w0, [c1]
   ret
```

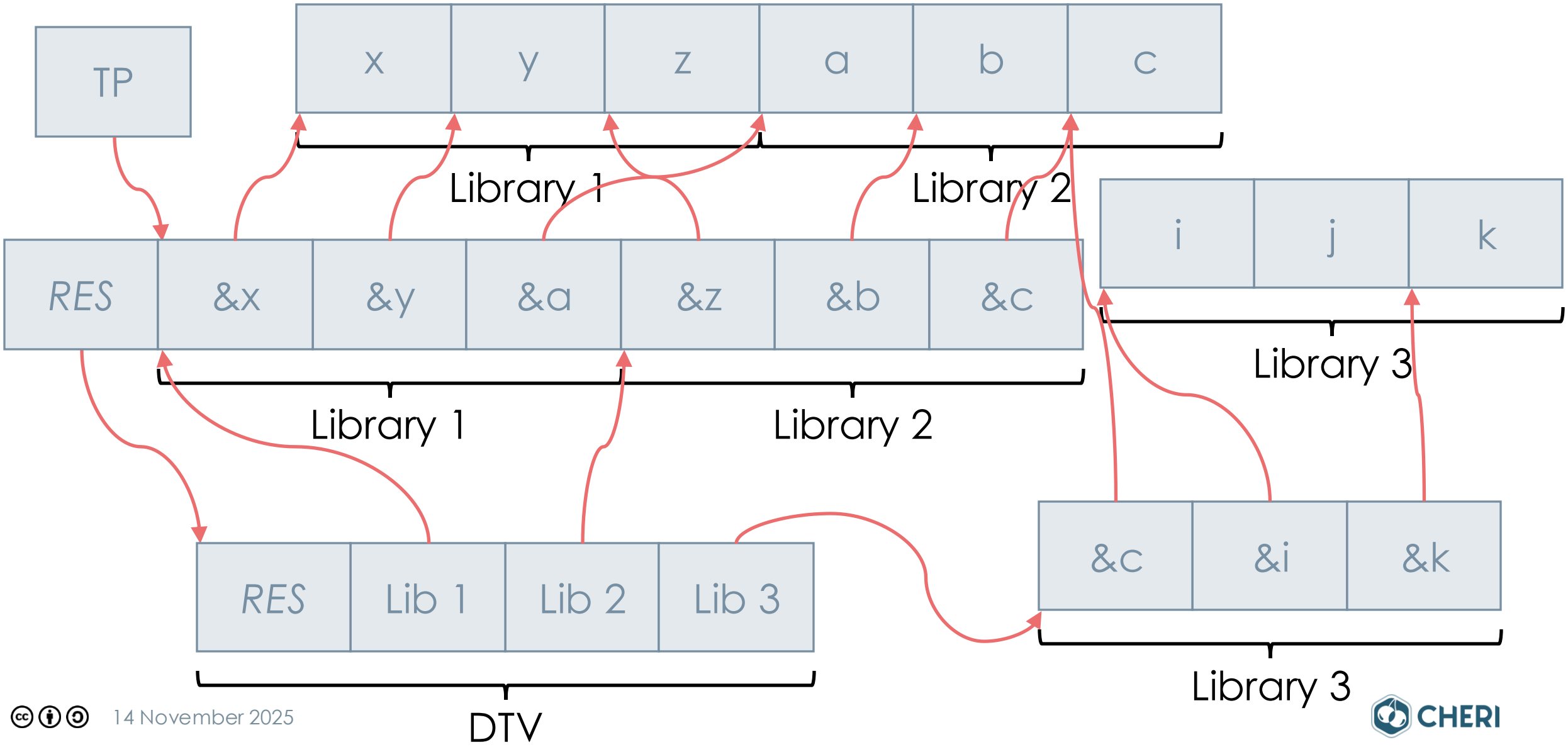
Load offset

Variable offset

dlopen() / DTV?

- As with TP, DTV now points to TGOTs (i.e. also add a level of indirection)
- TGOTs allocated separately for dynamically-loaded libraries just like the data itself
- Still a magic “__tls_get_addr()” function call (just implemented with extra pointer chasing)

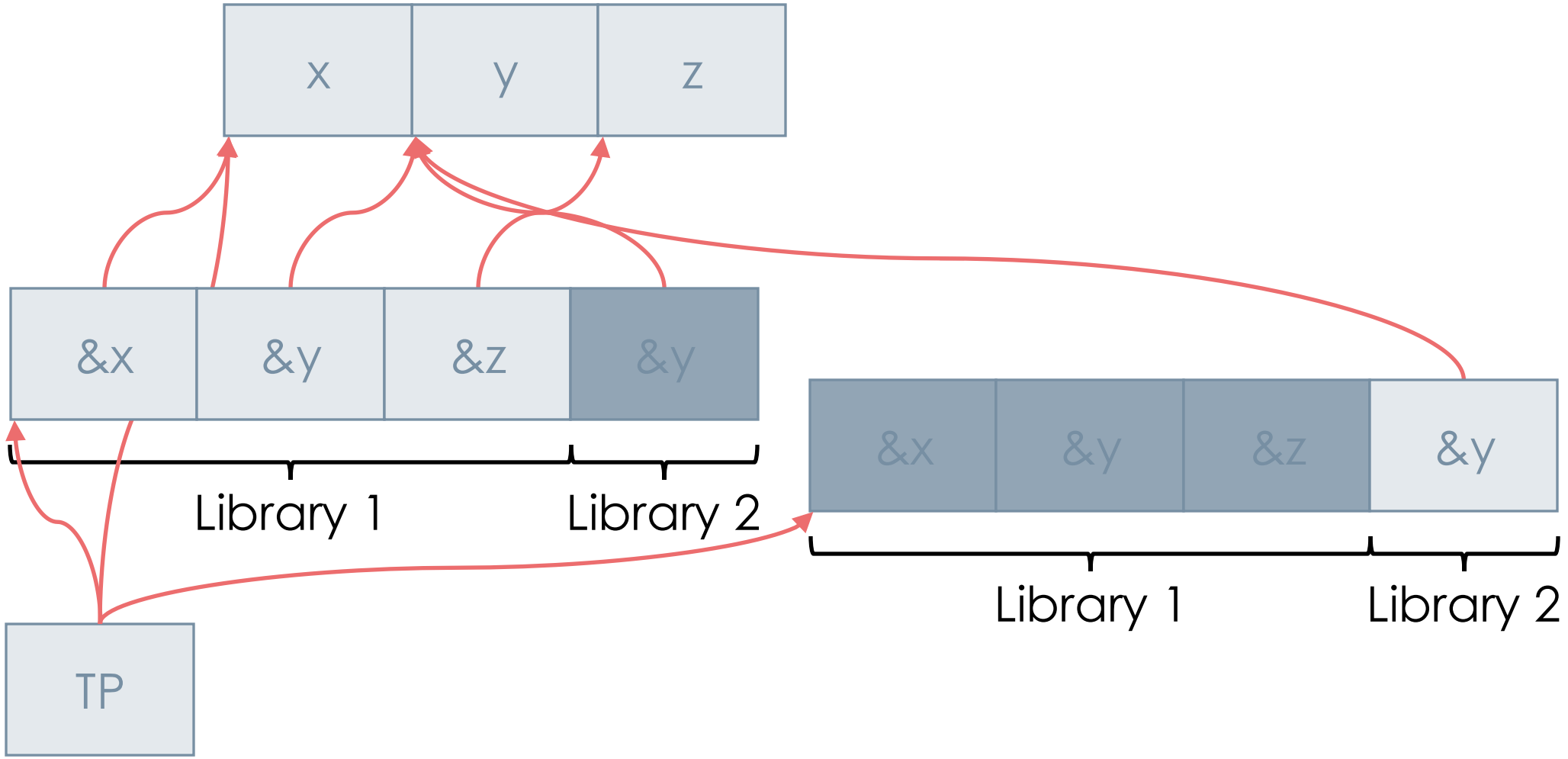
(Gratuitous) Diagram



Blue is Calming?

CHERI

Compartmentalisation



Future Work

- Implement compartmentalised TLS
- Evaluate performance impact (not expected to be significant)
- Coming to a future CheriBSD release!
  - Initial Morello version will implement both schemes at the same time to allow incremental transition

CHERITech'25 CONFERENCE

Manchester, UK

2025

14th November

cheri-alliance.org

Thank you!

Contact contact@cheri-alliance.org

Web www.cheri-alliance.org