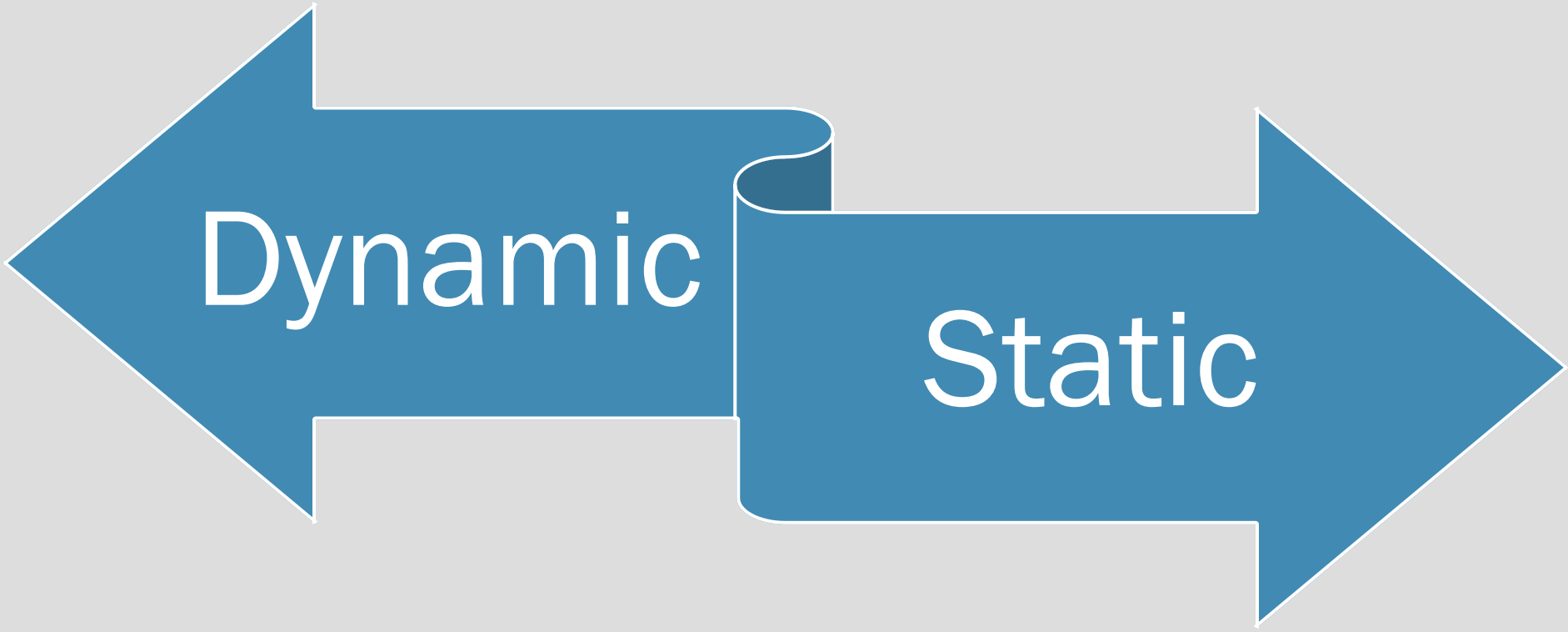
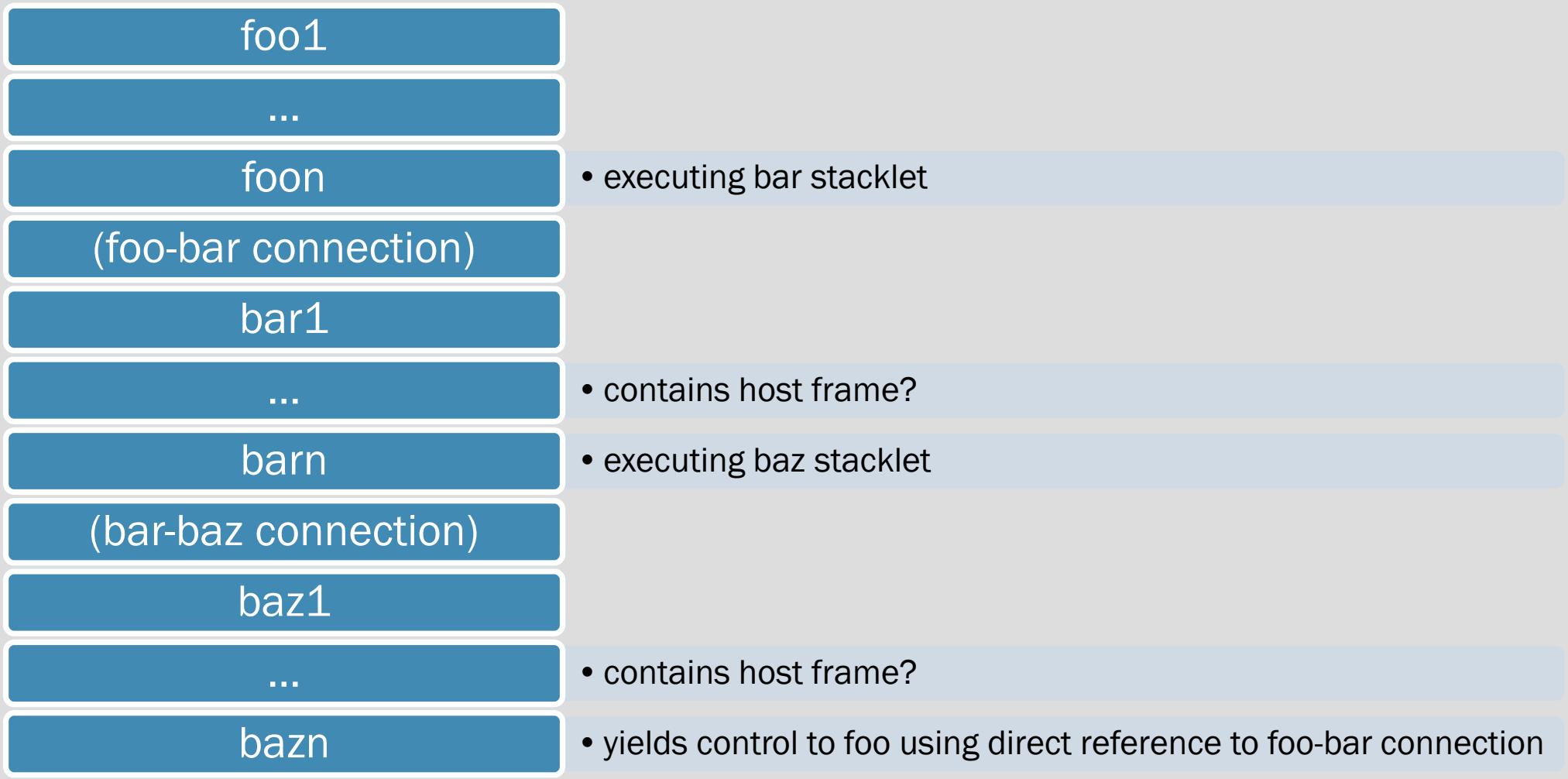
WHY TO LOCALIZE NON-LOCAL CONTROL

Ross Tate

Enforcement Strategies



Lexical Switching

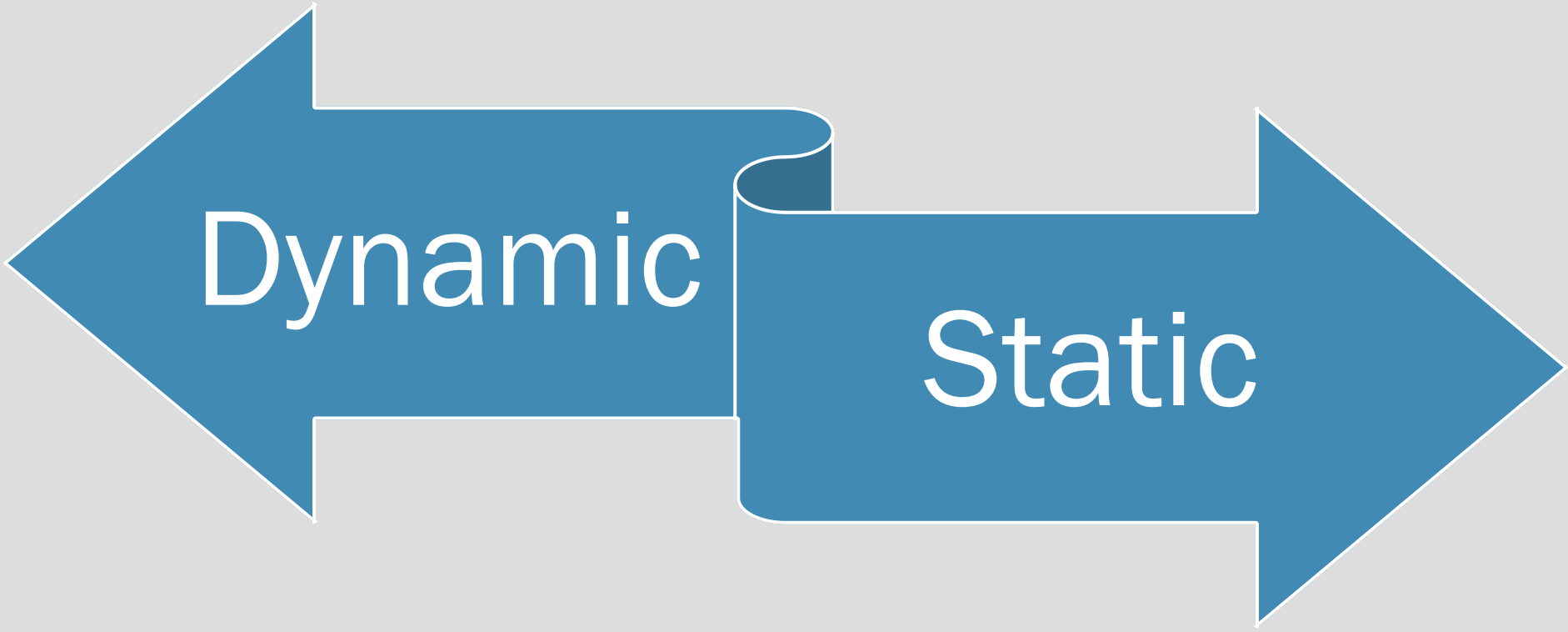


What to do when there is a host frame?

- Only suspend up to the host frame?
  - *Performance: every call from host into WebAssembly allocates a stack*
  - *Migration: wasm programs can come to depend on this functionality*
    - Difficult to change platform-specific host code to cross-platform wasm code
- Trap?
  - *Migration: difficult to change wasm host code to platform-specialized code*
- Corner cases?
  - *Interface types intertwines code from two sources (including host)*
  - *Different behavior for JavaScript frames versus C/C++/assembly frames?*

Host frames migrate between being foreign versus wasm

Enforcement Strategies



Effect Signatures

- (func $foo (param ti*) (result to*) (effect …))
  - *Specifics of … depends on other design choices (e.g. lexical vs. dynamic scope)*
  - *… at least indicates whether or not $foo can suspend*
- Caller of suspending $foo must either
  - *set up infrastructure for $foo to be able to suspend*
    - E.g. installs an effect handler or prompt or calls $foo on a new stacklet
  - *or declare itself to also be suspending (under same conditions as $foo)*
- No surprise suspends when calling effect-free wasm functions! (i.e. “opt-in” design)
  - *Make only effect-free wasm functions accessible to host language (e.g. C or JS)*
    - Guarantees host frames will not be captured

Faulty Comparison

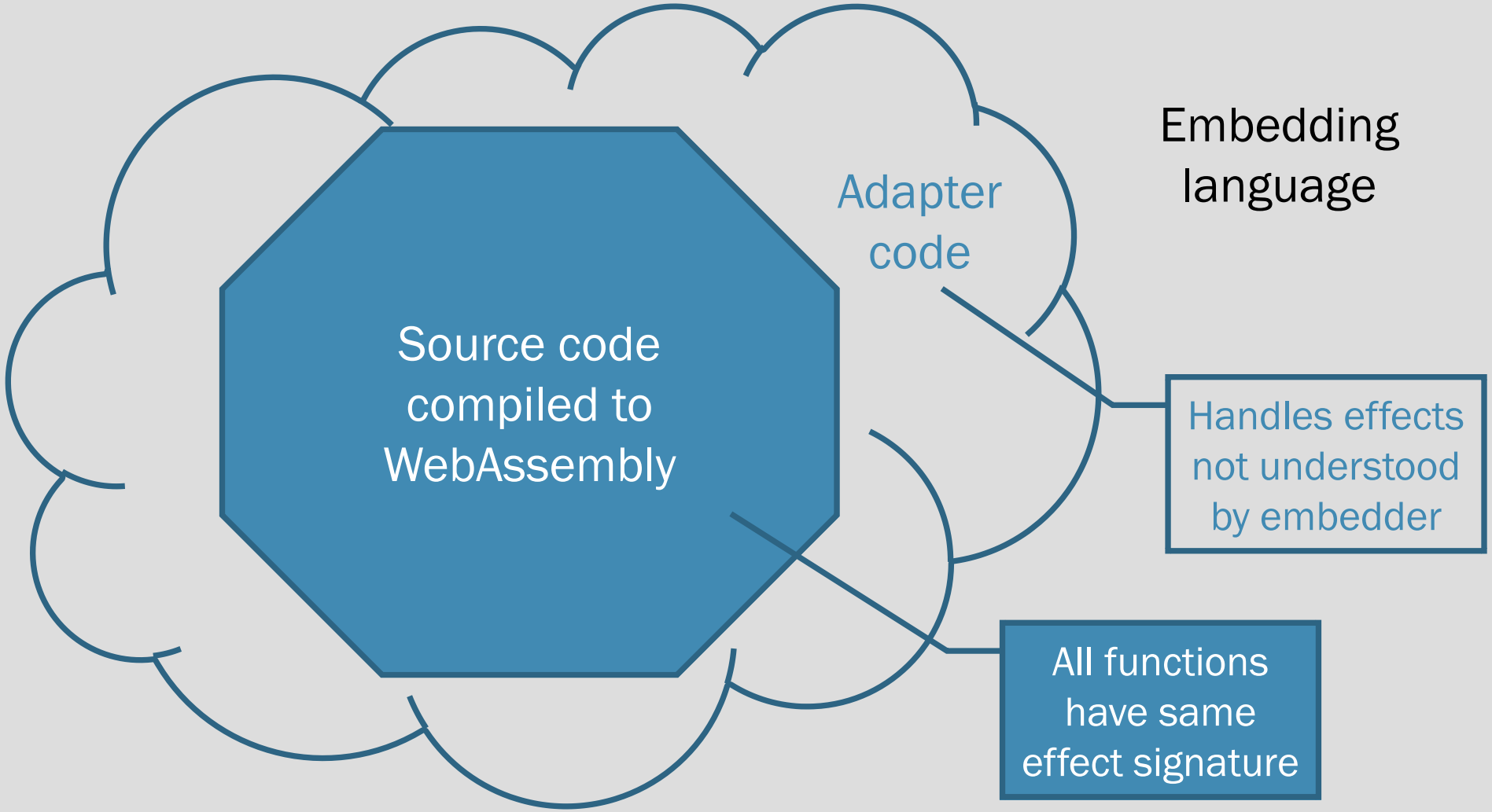
Java Exceptions

- Many exception types within module
- No guarantees (RuntimeException)
- Written by hand

Wasm Effect Signatures

- One effect type in typical module
- Strong guarantees
- Procedurally generated

Easy to generate (preliminary estimate)



Simpler in the long run

No unknown/foreign effects

- No need for handle_all (i.e. catch_all)
- No need for eventref (i.e. exnref)
- No need for dynamic-wind (i.e. unwind)

Program invariants automatically respected

- Program can easily keep track of all its stack frames (for linear-memory GC root marking)
- Stacks cannot be unexpectedly duplicated inside critical regions
- Shadow stacks stay aligned with wasm stacks

Enforcement Strategies