Weak Memory Model for Wasm

Status update

Andreas Rossberg

Dfinity

Conrad Watt

Cambridge University

Background

Thread proposal brings low-level shared state concurrency to Wasm

Exposes weak memory semantics

Need semantic model

Research problem!

Delicate interop with JS SharedArrayBuffer

Research

Collaborating with Conrad Watt

PhD student of Peter Sewell, Cambridge

Weak memory model for Wasm is his thesis

Does most of the work, I’m just throwing twigs

Why is it hard?

Various aspects still an open research problem

Incredibly subtle and difficult to validate

Models for languages like C++ are fairly recent

…and have a range of deficiencies

Various novel aspects in Wasm

…both relative to C++ and relative to ES

Integration with operational semantics

…overlay reduction relation with axiomatic constraints

…integration with syntactic soundness proof

Not in any prior model

Growing memory

…need to order access failures!

Multi-language execution via host functions

…axiomatise legal effects on store,
store extension, thread creation

…reentrancy and interleaving with own execution

Not in ES

Growing memory

Host functions

Semantics of thread creation

Formal semantics of memory creation

Need to extend over SAB model

Also, some minor fixes and tweaks to SAB model

Not in C++

Growing memory

Host functions

Fully defined behaviour!

…no throw-up-hands-and-catch-fire

Mixed-size atomic memory access

…first proper investigation in recent POPL paper

…hardware is incredibly weak

…ES model is even weaker

```
i32.atomic.load8 2  ||
i32.atomic.load8 2  ||  i32.atomic.store16 2 1
i32.atomic.load8 2  ||
```

Each load could individually see 0 or 1

Even 1,0,1 is legal in ES model

```
memory 1

i32.store 0x10000 1  ||  memory.grow 1
i32.store 0x10000 2  ||
```

Non-atomic stores,
both are OOB before but not after grow

Grow has to be sequentially consistent with any access failure

…cannot reorder stores

More fun with tearing store-at-boundaries vs grow…

NB: ES spec probably needs changes around detaching

Status

Have a sketch of a model

Need to work through meta-theory

Need to integrate it with Wasm spec

Need to describe ES interop

Need to tweak ES spec

Summary

Challenging technical problem

But we think we have a good handle on it now

Still have to work out the details

ETA later this year
(thanks Spectre for buying us more time :))

ES requires some tweaks, too