

How to not make bugs (in JSC), and the future of 32-bits

WebKit Contributors Meeting 2025
28-29 October 2025

Justin Michaud
jmichaud@igalia.com

Who Am I?



Outline

- 32-bit port challenges
- 32-bit JSValues
- Bugs!



32-bit port

- Lack of registers
- JSValues / i64 can't fit in register
- Small VA space



What is a JSValue?

- Double
- Int32
- Boolean
- Undefined, null, empty
- JSCell*



What is a JSValue?

```
0x0000_PPPP_PPPP_PPPP_PPPP_PPPP_PPPP_PPPP_PPPP <- Ptr  
0x0002_DDDD_DDDD_DDDD_DDDD_DDDD_DDDD_DDDD_DDDD <- Double (+0x0002)  
0xFFFC_XXXX_XXXX_XXXX_XXXX_XXXX_XXXX_XXXX  
0xFFFE_XXXX_XXXX_XXXX_XXXX_XXXX_XXXX_XXXX
```

```
^ ----- QNaN ^  
Payload (nonzero) ----- |
```



What is a 32-bit JSValue?

XXXX_XXXX_XXXX_XXXX YYYYY_YYYY_YYYY_YYYY

^----- Tag ^
Payload -----|



What is a 32-bit 32-bit JSValue?

```
... XXXX_AABC
    ^----- Boxed double/int
      ^---- Is JSCell*?
        ^--- Is SMI?
```



Why?

- Avoid different code paths vs. 64-bit
- Concurrent GC
- 64-bit mini mode



I64s

- Static register pairs



GC Bugs



GC Bugs

- Conservative stack scanning



GC Bugs

```
Structure* oldStructure = WTF::opaque(baseObject->structure())
Structure* newStructure = WTF::opaque(entry->m_newStructureID)

...

ensureStillAliveHere(oldStructure);
ensureStillAliveHere(newStructure);
```



Generational Barriers

```
Old { foo: | _ | }  
          |-----> Young { }
```



Concurrent Barriers

```
foo = {}  
foo.a = { }  
// foo is marked  
foo.b = { }  
// sweep
```



32-bit barriers

- Add barriers in more places? (ex: Int32Shape typed arrays)
- Unbox ints in more places?
- Use incremental, non-generational, concurrent STW collector instead?



A speculation cliff



What is Int8Array?

```
new Int8Array(9)
```

```
Array(9)
```



Make predictions now



Perf

Phase 1	Phase 2	Slowdown
U	U	1
A	A	1.72
A	U	22.8
U	A	6.3

V8

Phase 1	Phase 2	Slowdown
U	U	1
A	A	2
A	U	2.79
U	A	2.38



Fast

```
D@303:<!1:->    CheckInBounds(Int32:D@201, KnownInt32:D@283)
D@247:< 7:->    GetByVal(KnownCell:D@69, Int32:Kill:D@201)
  D@83:<!0:->    CheckStructure(Check:Cell:D@247, [%Ap:Int8Arr
  D@77:< 2:->    GetIndexedPropertyStorage(KnownCell:D@247, In
D@208:< 2:->    GetUndetachedTypeArrayLength(KnownCell:D@247,
D@258:<!1:->    CheckInBounds(Int32:D@137, KnownInt32:D@208)
D@251:<!2:->    *GetByVal*(KnownCell:D@247, Int32:D@137, Chec
D@254:<!0:->    Branch(Int32:Kill:D@251)
```



Comparison against other languages

Language	Speedup
JS	1
CPP	3.5x Faster
Rust	2.6x Faster
Java	1.6x faster
Swift	3.8x Slower
Pypy	5x Slower
Python	71x Slower

How to not make bugs (in JSC), and the future of 32-bits

Igalia, WebKit Contributors Meeting 2025



But seriously, JS?



Questions?



Thanks!



