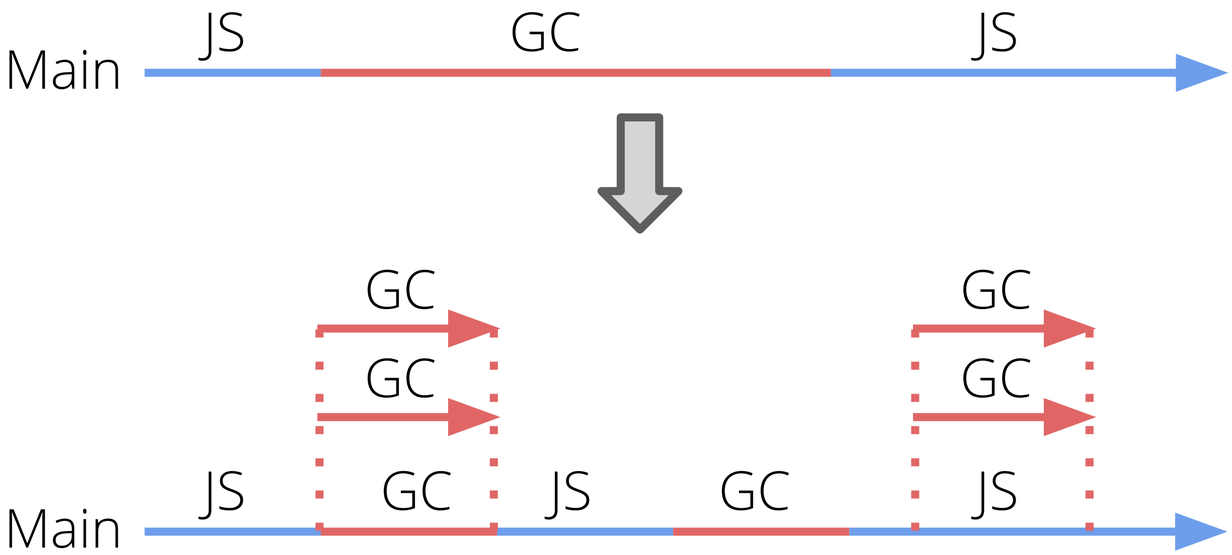
V8 Garbage Collection

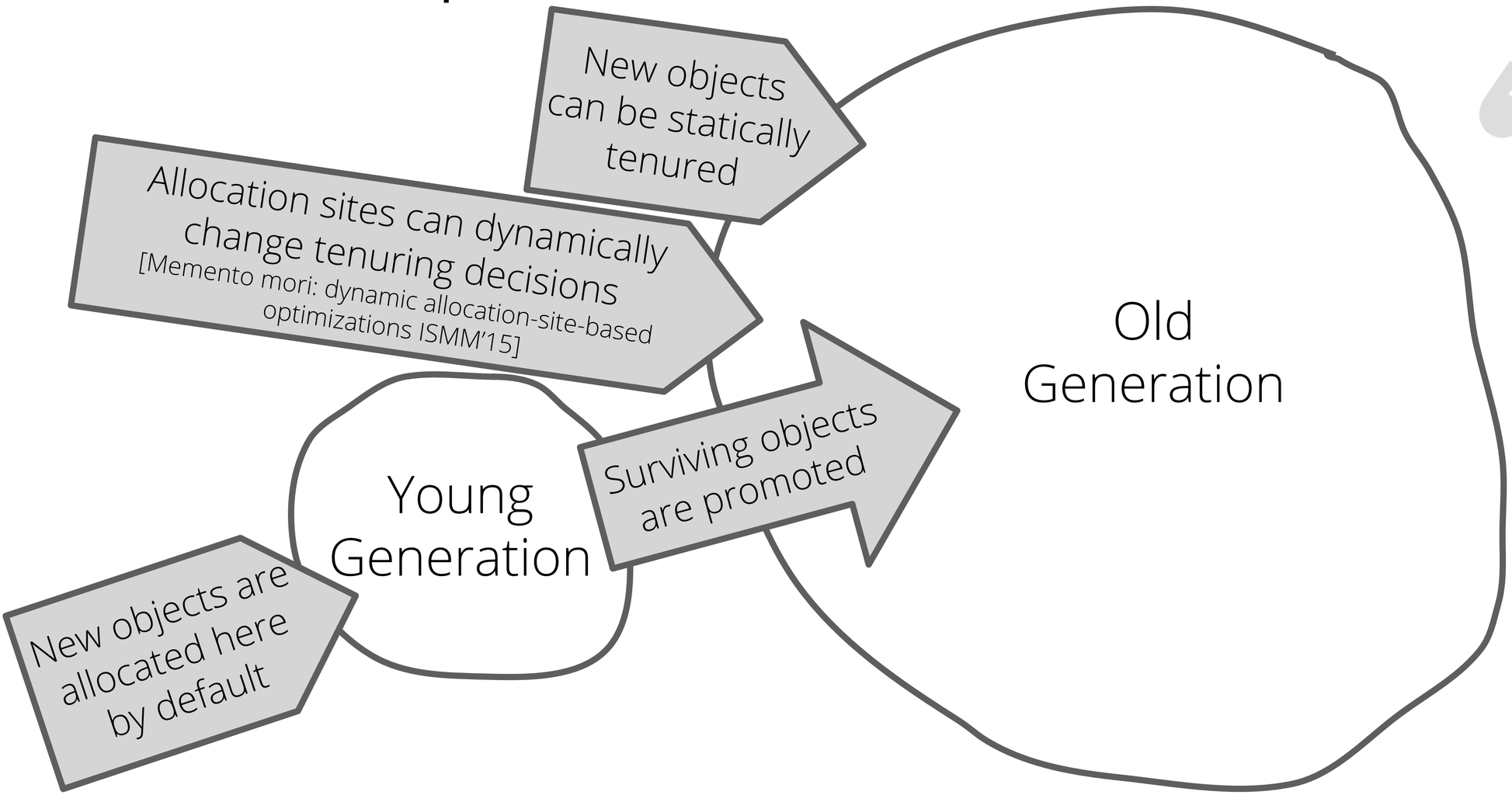
Hannes Payer
Google|Chrome|V8
https://research.google.com/pubs/HannesPayer.html

V8 Orinoco

A generational, moving, mostly parallel and concurrent garbage collector with incremental fallback.



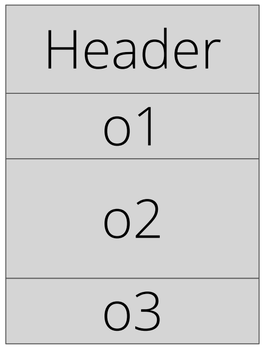
V8 Heap Architecture



"Most objects will die young."

V8 Heap Architecture

Regular Pages



- 256K right now
- Meta-data page header: marking bitmap, remembered set, etc.
- Fast page lookup mechanism from objects
- Page-based parallelization

Large Objects

- Arbitrary size
- Young or old generation
- Slow page lookup mechanism from within large objects

V8 Orinoco Features

Young Generation GC

- Up to 16M
- Semi-space
- Parallel Scavenger
- Alternative minor Mark-Compact --minor-mc

Full GC

- Old & Young Generation
- Concurrent, parallel & incremental marking
- Concurrent, parallel & incremental sweeping
- Parallel compaction

Young Generation Garbage Collector

Scavenger

- Single-pass over the young generation
- Fast when most objects die young
- Slow when many objects survive (99%tile)

Minor Mark-Compact

- Two passes over the young generation
- Copy-free promotion
- Too slow on the common cases
- Faster on the higher %tiles

Young Generation Garbage Collector

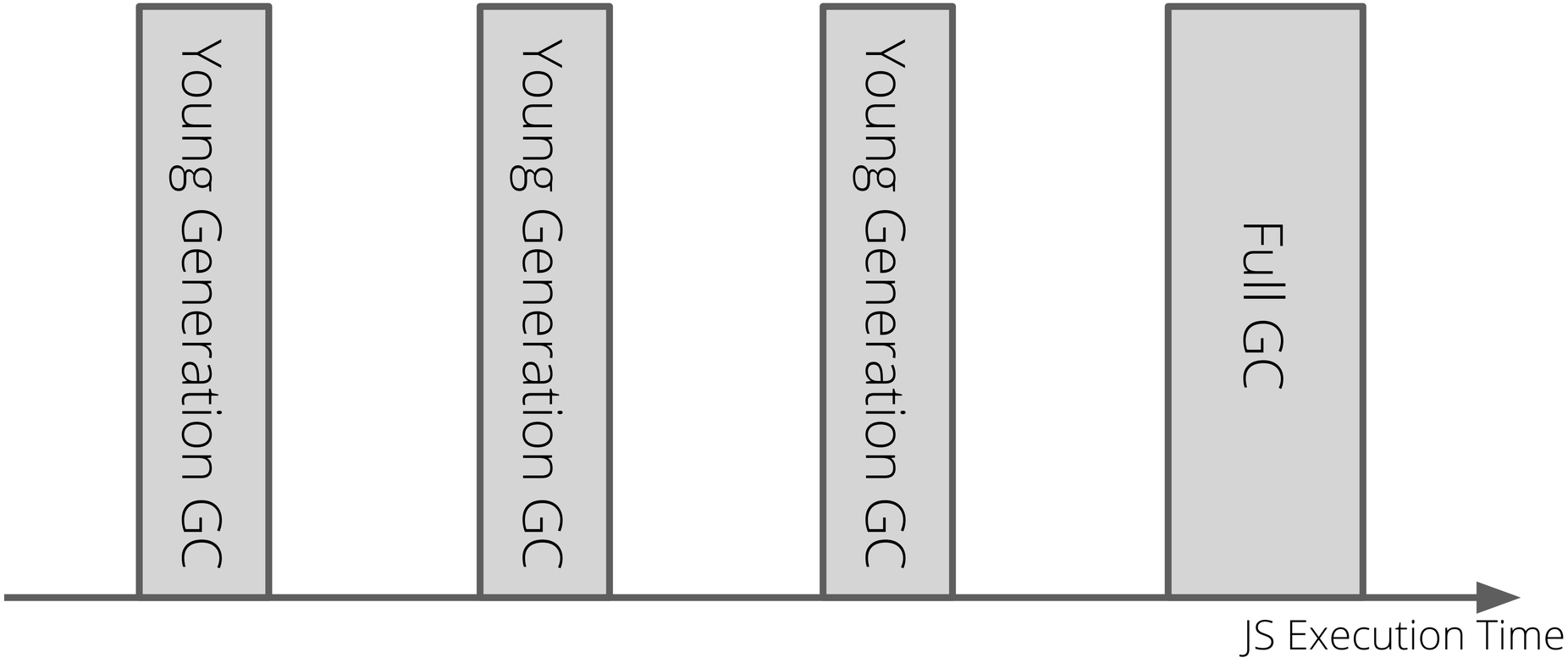
Scavenger

- Single-pass over the young generation
- Fast when most objects die young
- Slow when many objects survive (99%tile)

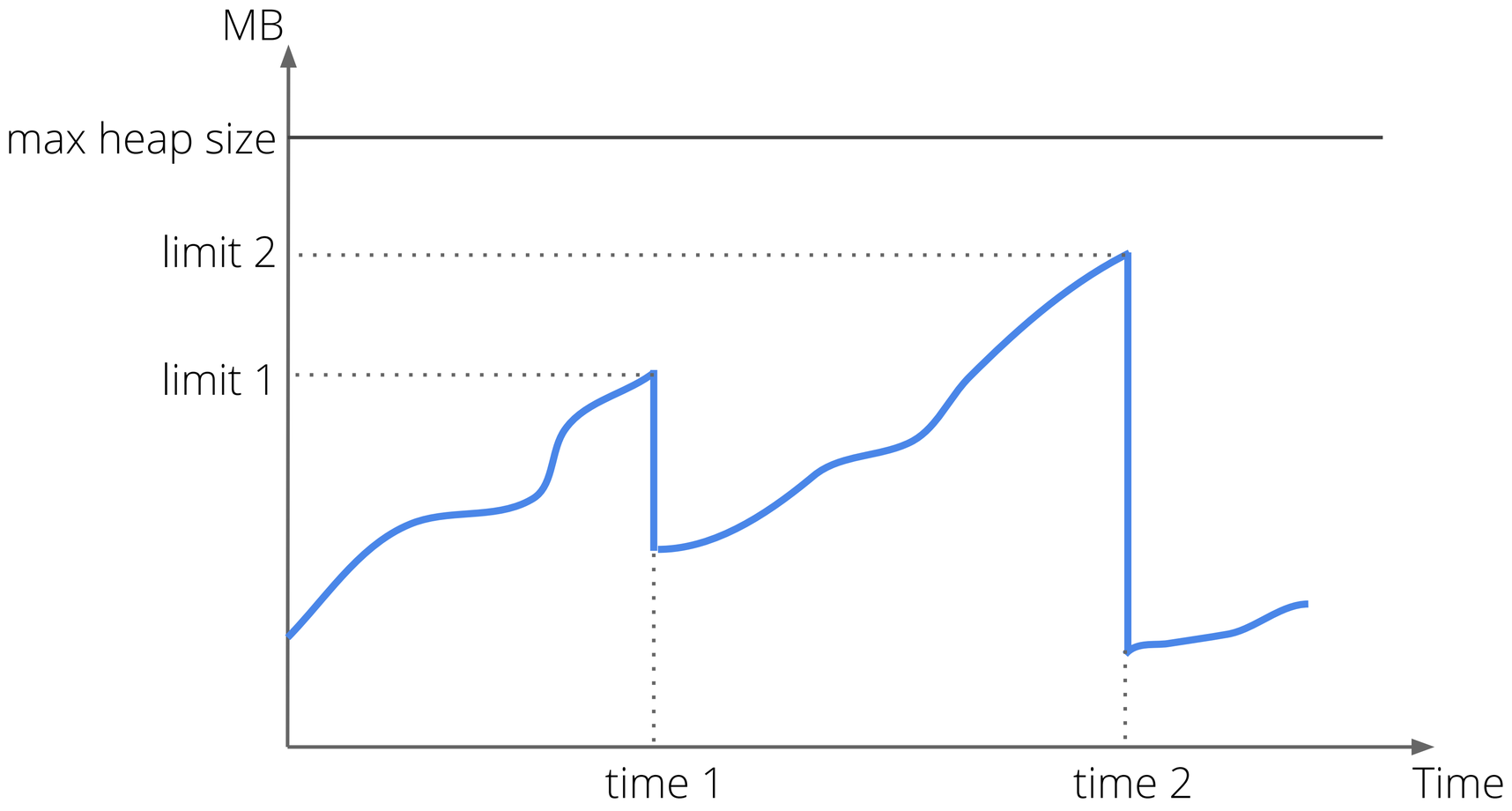
Minor Mark-Compact

- Two passes over the young generation
- Copy-free promotion
- Too slow on the common cases
- Faster on the higher %tiles

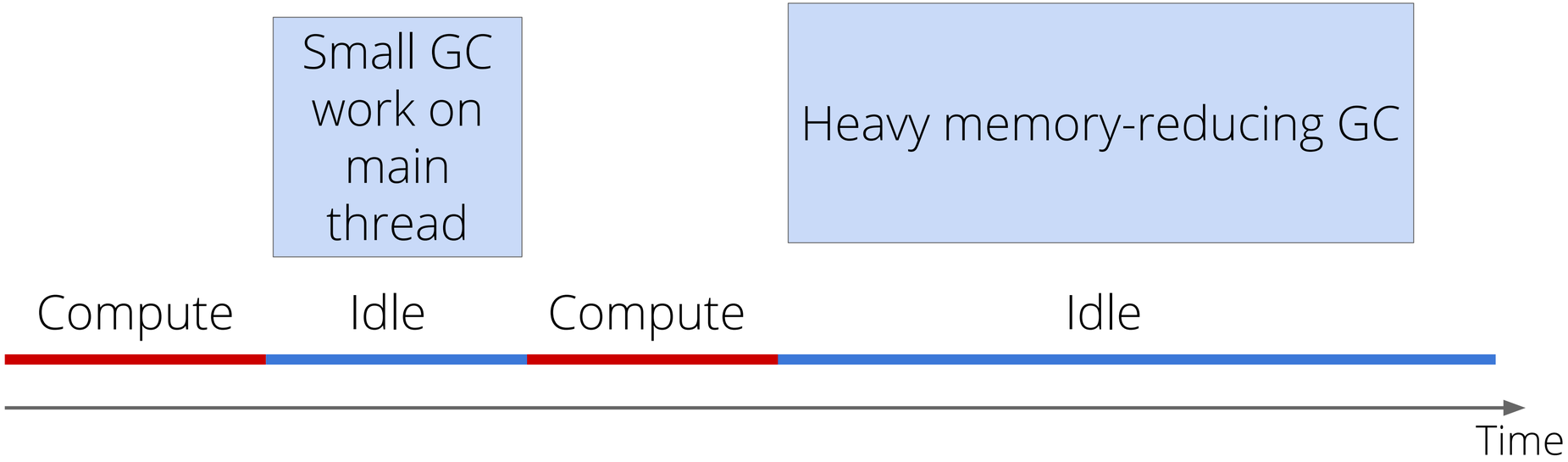
V8 Garbage Collection Events



Garbage Collection Scheduling

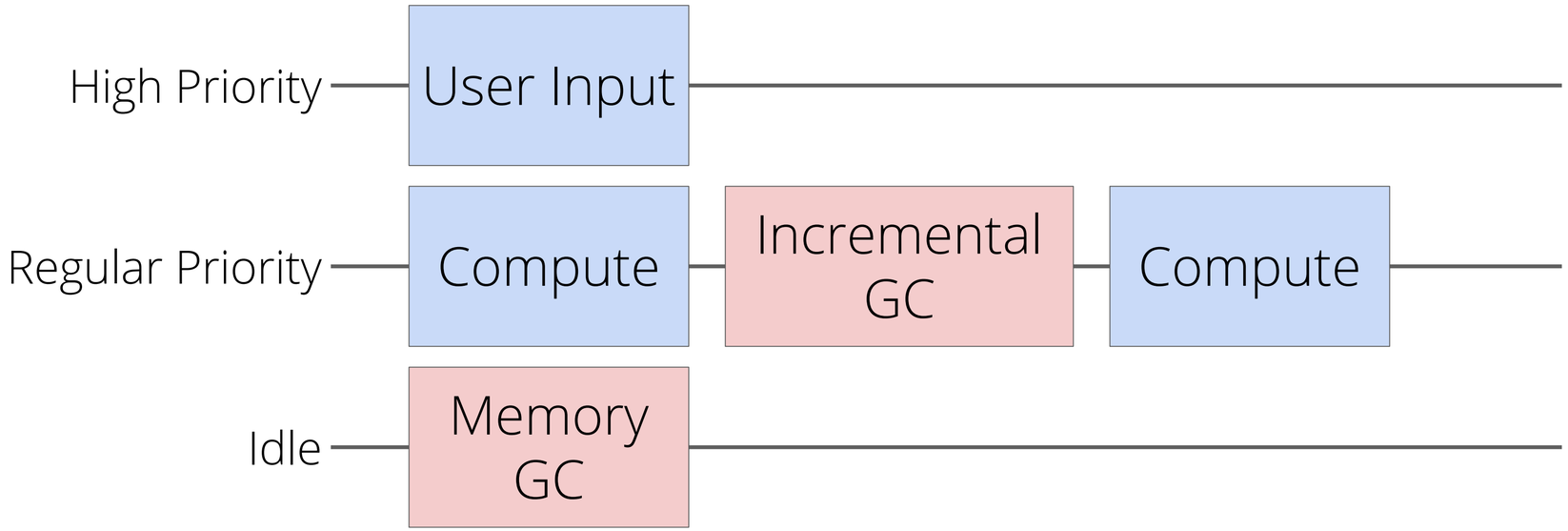


Garbage Collection Scheduling



Idle Time Garbage Collection Scheduling
[PLDI'16, Communications ACM 59(10)]

Garbage Collection Scheduling



Priority queues of the main thread task scheduler

Reduce Queuing time of tasks.

Concurrent Marking is a solved problem!

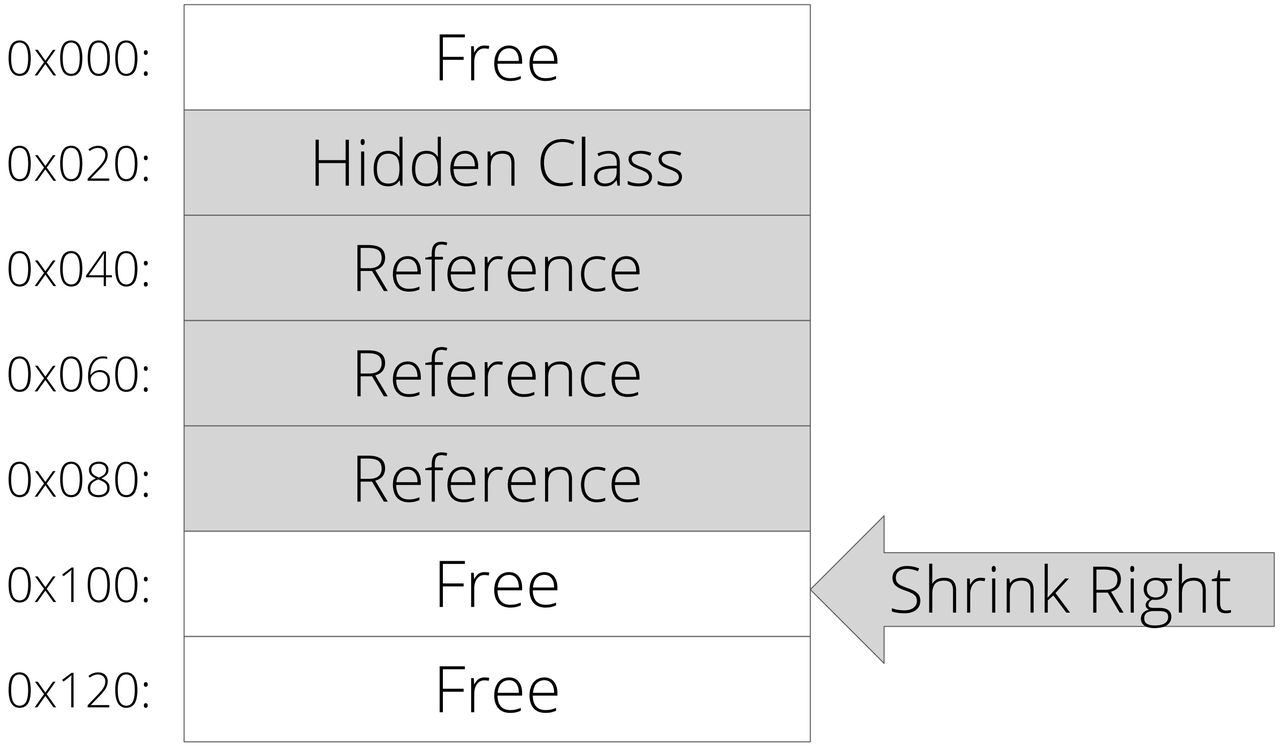
Really!?

Concurrent Marking of Shape-Changing Objects

0x000:	Free
0x020:	Hidden Class
0x040:	Reference
0x060:	Reference
0x080:	Reference
0x100:	Reference
0x120:	Free

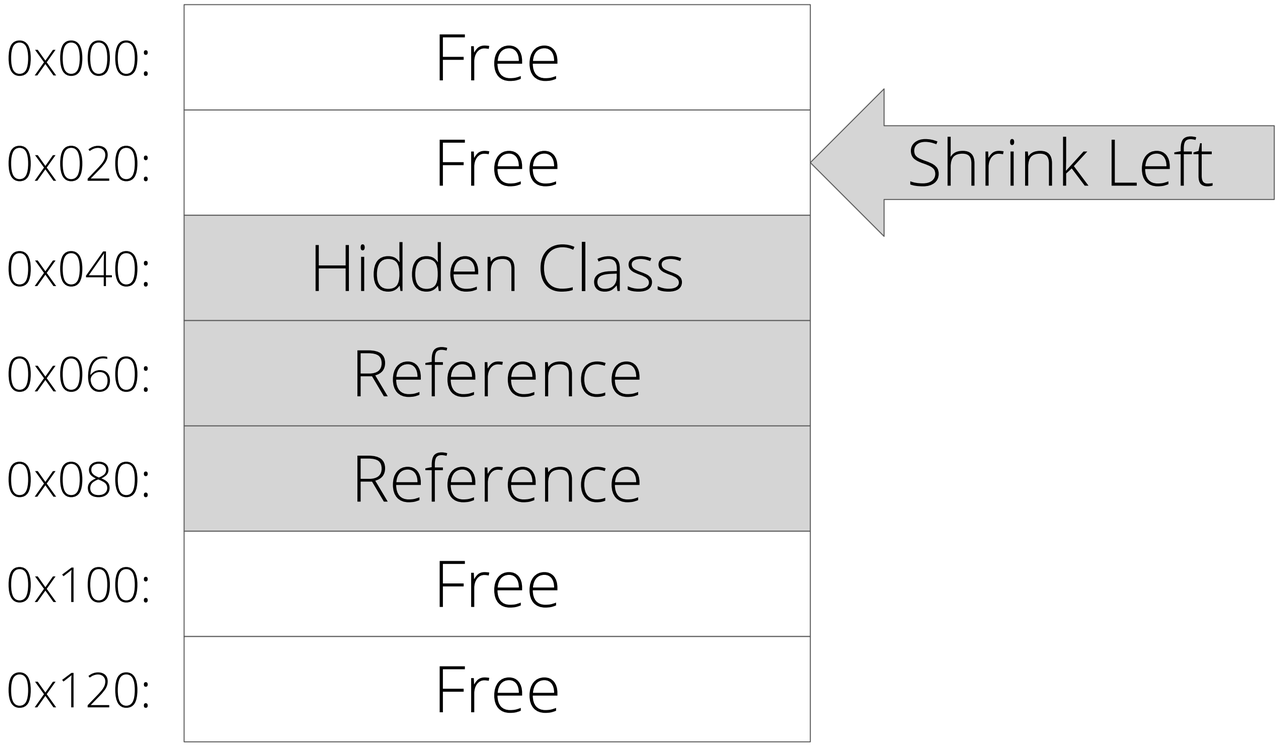
Concurrent Marking of Shape-Changing Objects [ISMM'19]

Concurrent Marking of Shape-Changing Objects



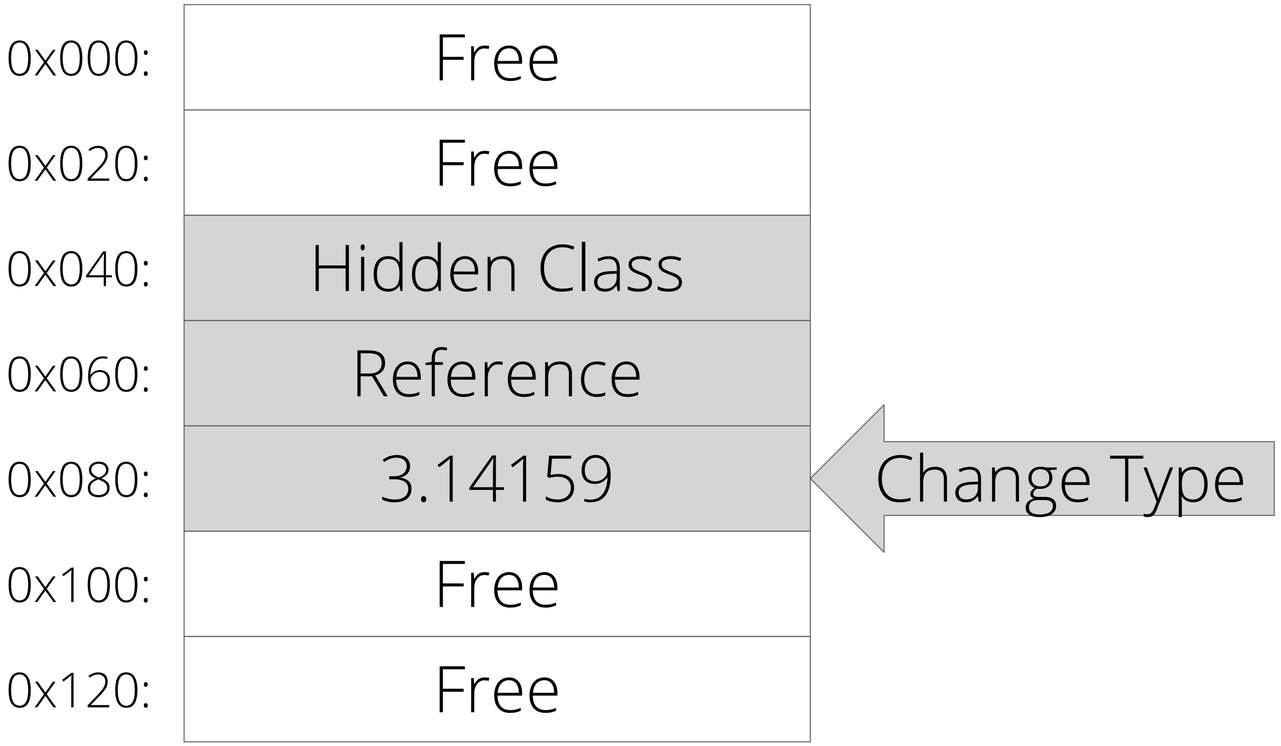
Concurrent Marking of Shape-Changing Objects [ISMM'19]

Concurrent Marking of Shape-Changing Objects



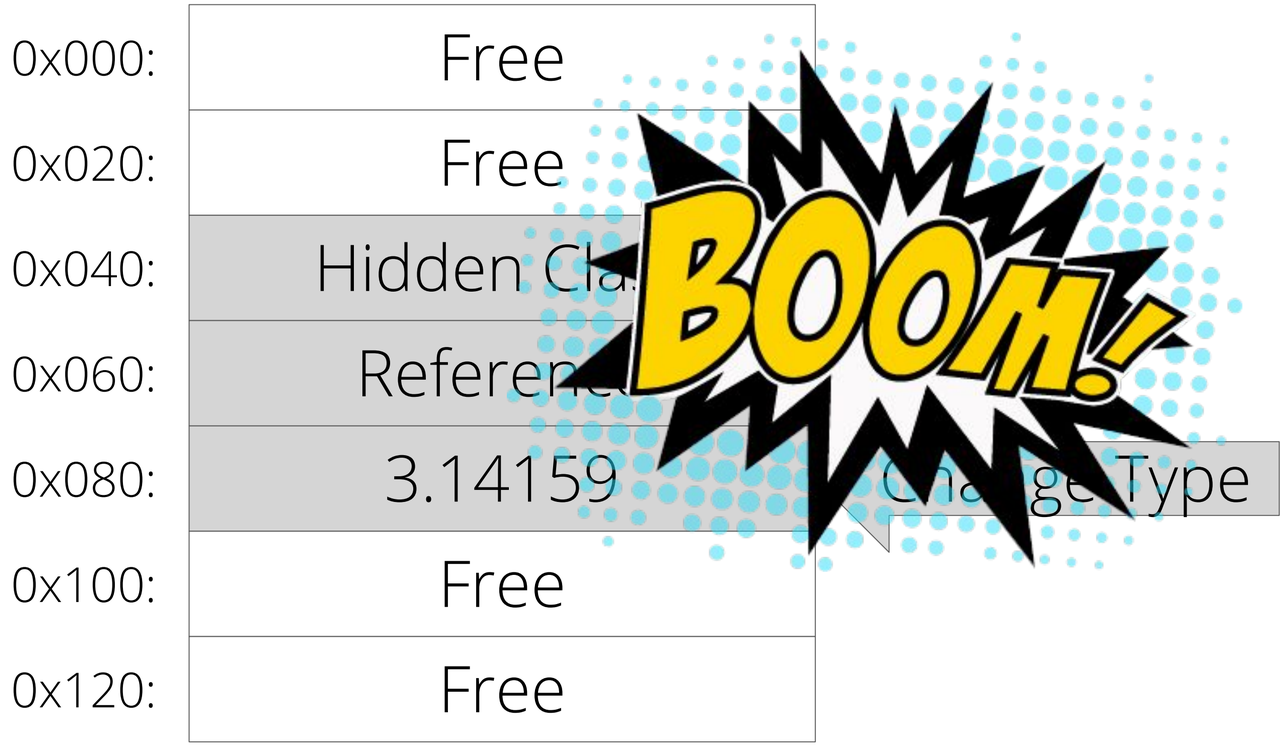
Concurrent Marking of Shape-Changing Objects [ISMM'19]

Concurrent Marking of Shape-Changing Objects



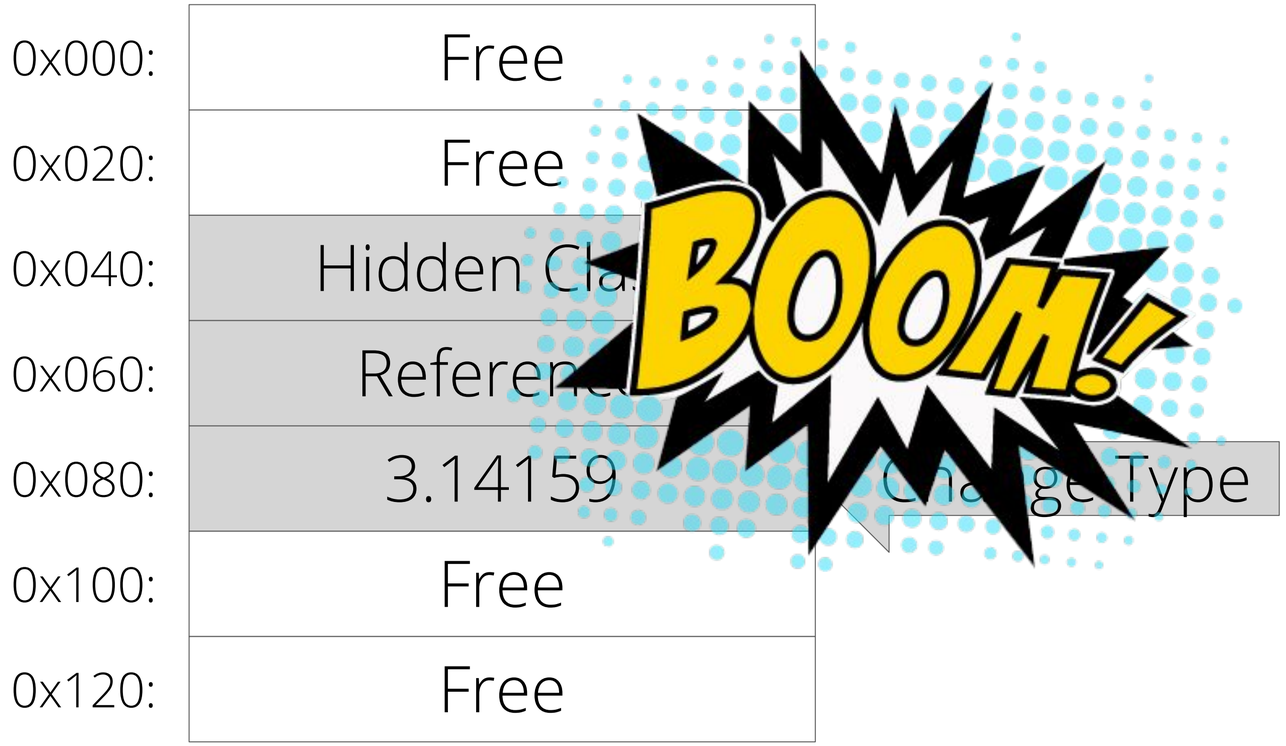
Concurrent Marking of Shape-Changing Objects [ISMM'19]

Concurrent Marking of Shape-Changing Objects



Concurrent Marking of Shape-Changing Objects [ISMM'19]

Concurrent Marking of Shape-Changing Objects



Tomorrow 14:45 - 15:15
Room 106A

Concurrent Marking of Shape-Changing Objects [ISMM'19]

Heap/Garbage Collector Interface

1. Memory allocator: Bump-pointer

2. Write Barriers:
   - Old to young generation objects
   - References that point to objects which may be compacted
   - Marking (Dijkstra-style):

```
store obj.slot[x] = p

if (load p.color) == white:
  store p.color = grey
  push p
```

Automatically added by the Compiler and Runtime

Write Barrier Elimination: Allocation Folding Based on Dominance [ISMM'15]

Cross-Component Garbage Collection in Chrome

Blink

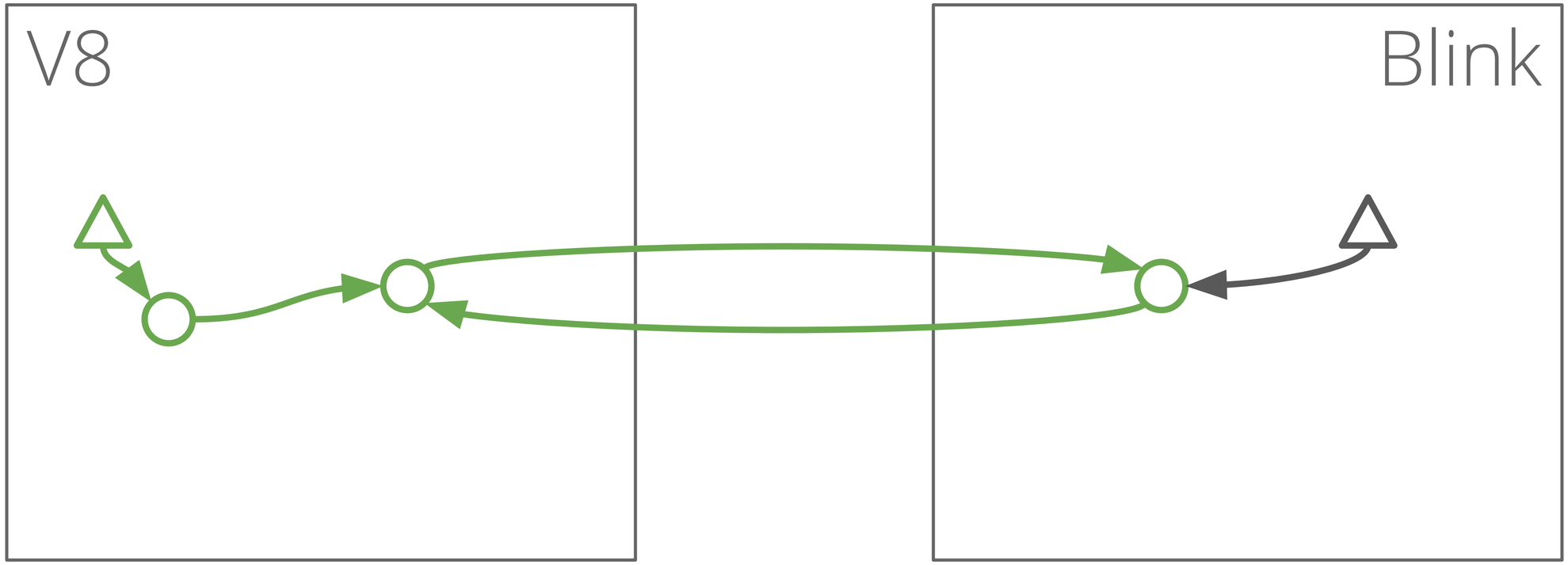
- Chrome’s web browser engine which embeds V8
- JavaScript is used to dynamically modify the DOM
- V8 objects can reference objects on the Blink heap and vice-versa
- Most of the C++ Blink heap is managed by the precise & conservative Mark-Sweep-Compact Blink garbage collector Oilpan for C++ objects

Challenge:

How do we collect the full transitive closure of objects spanning both components?

No dangling pointers

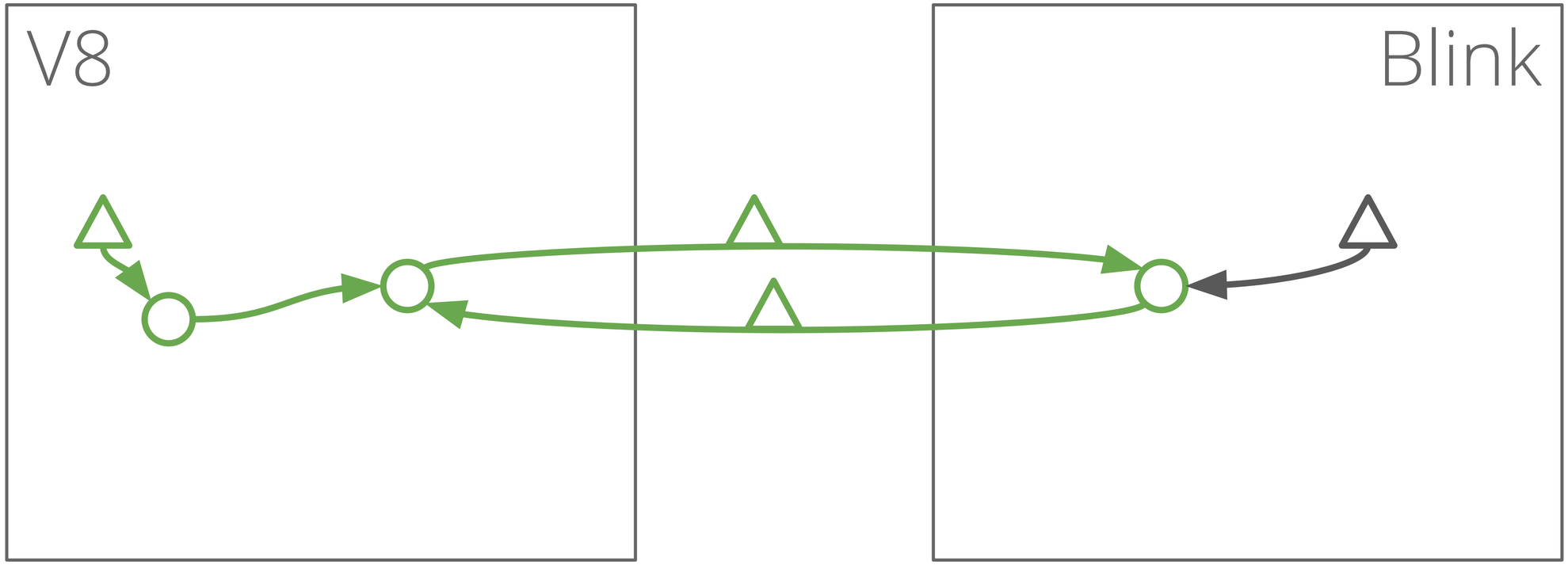
No memory leaks



Root reference

Weak reference

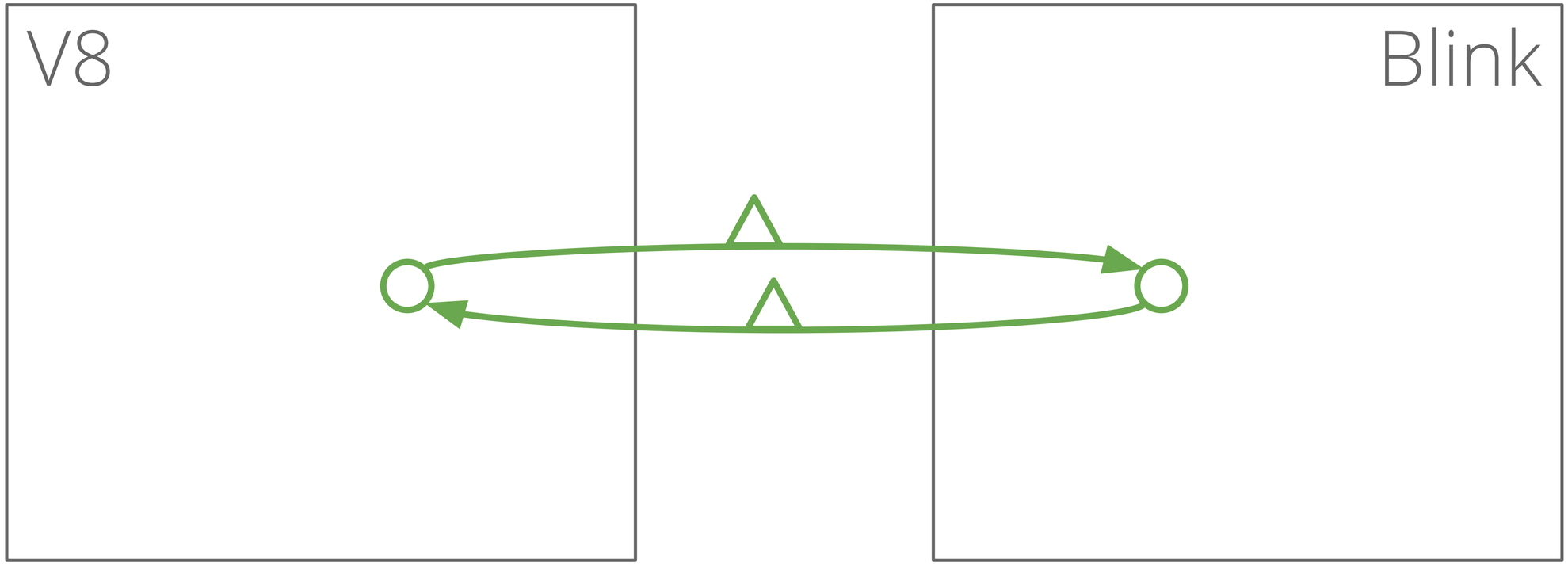
Traced reference



Root reference

Weak reference

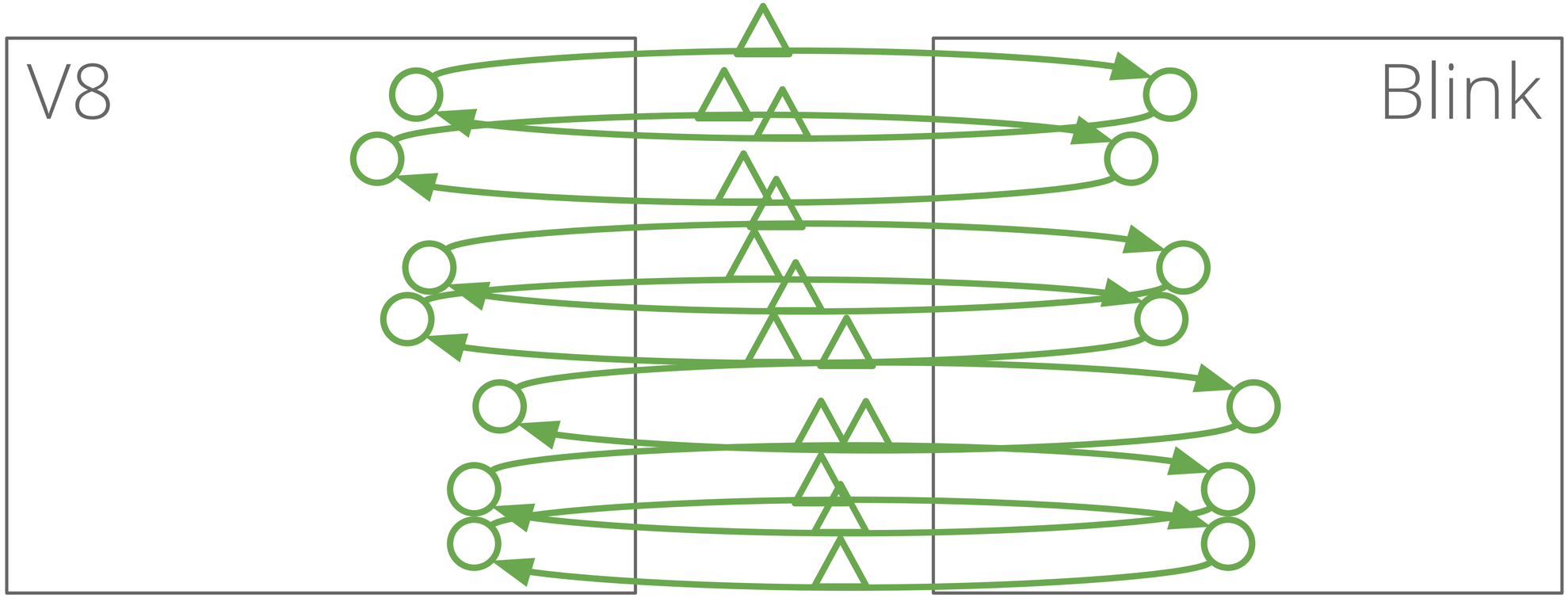
Traced reference



Root reference

Weak reference

Traced reference



Root reference

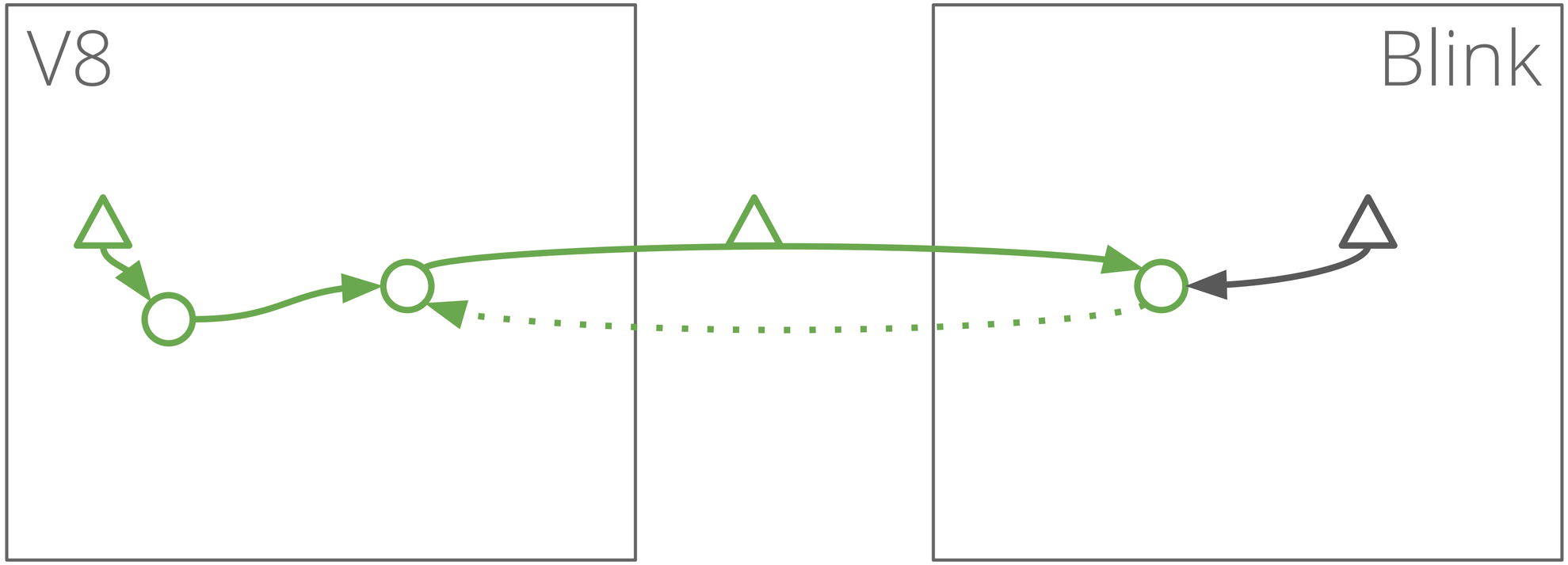
Weak reference

Traced reference

Root reference

Weak reference

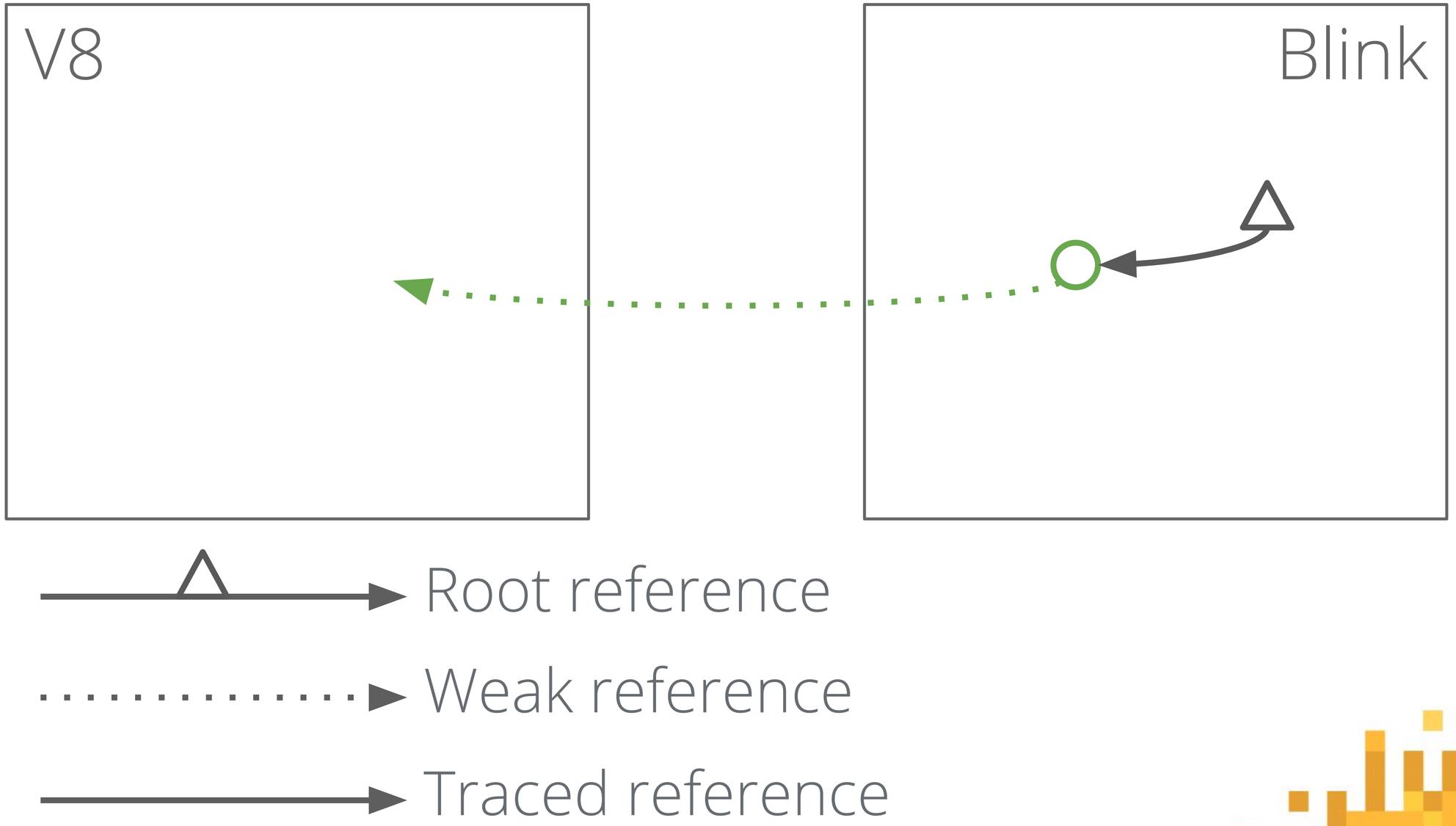
Traced reference

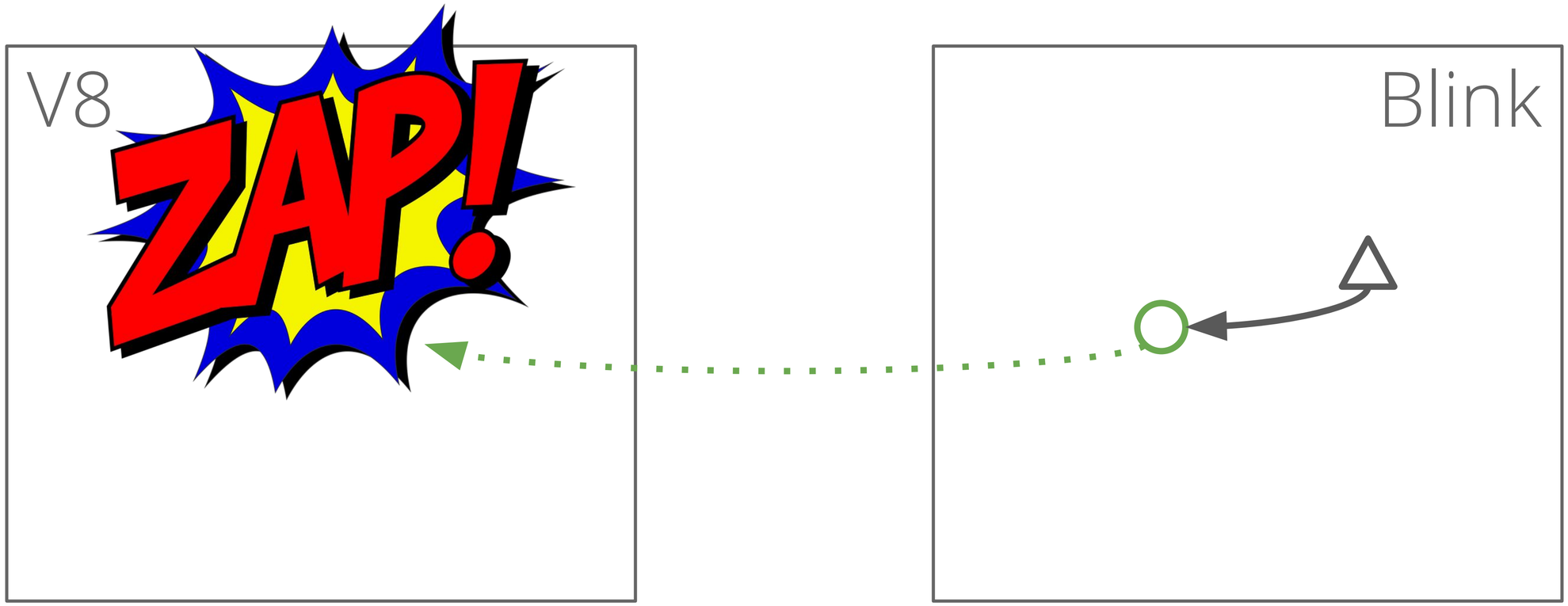


Root reference

Weak reference

Traced reference

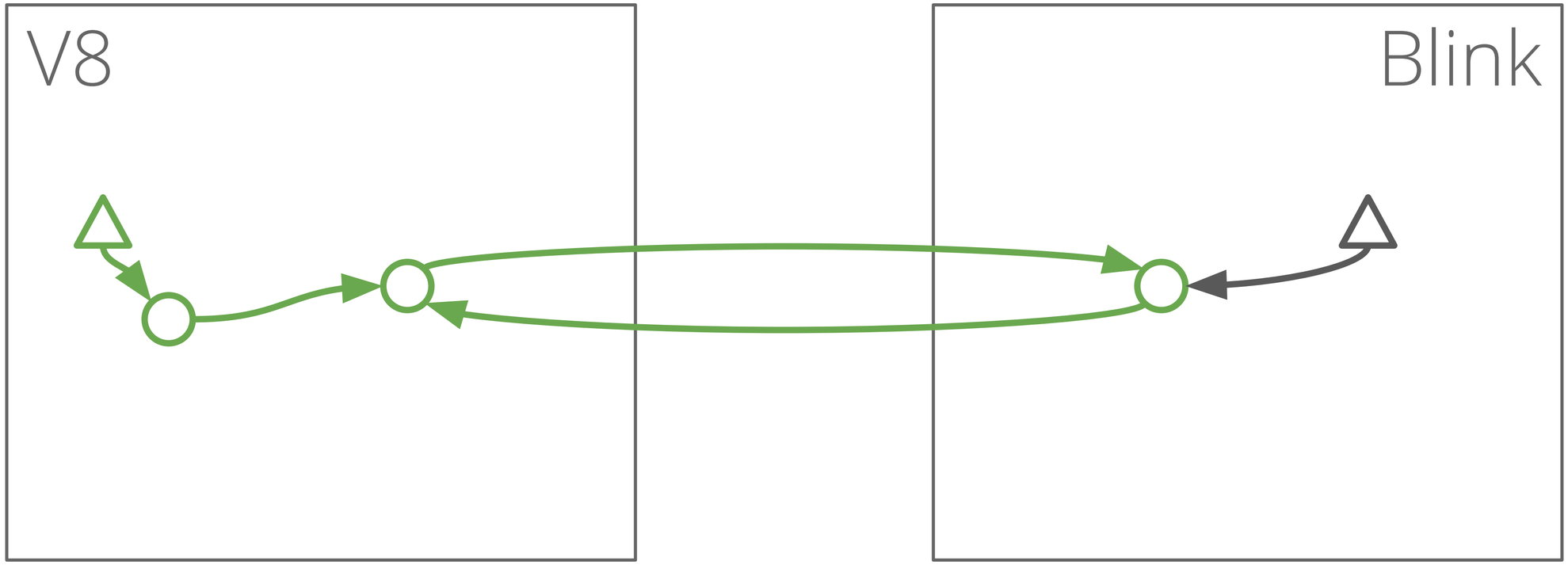

V8
Blink
Root reference
Weak reference
Traced reference



Root reference

Weak reference

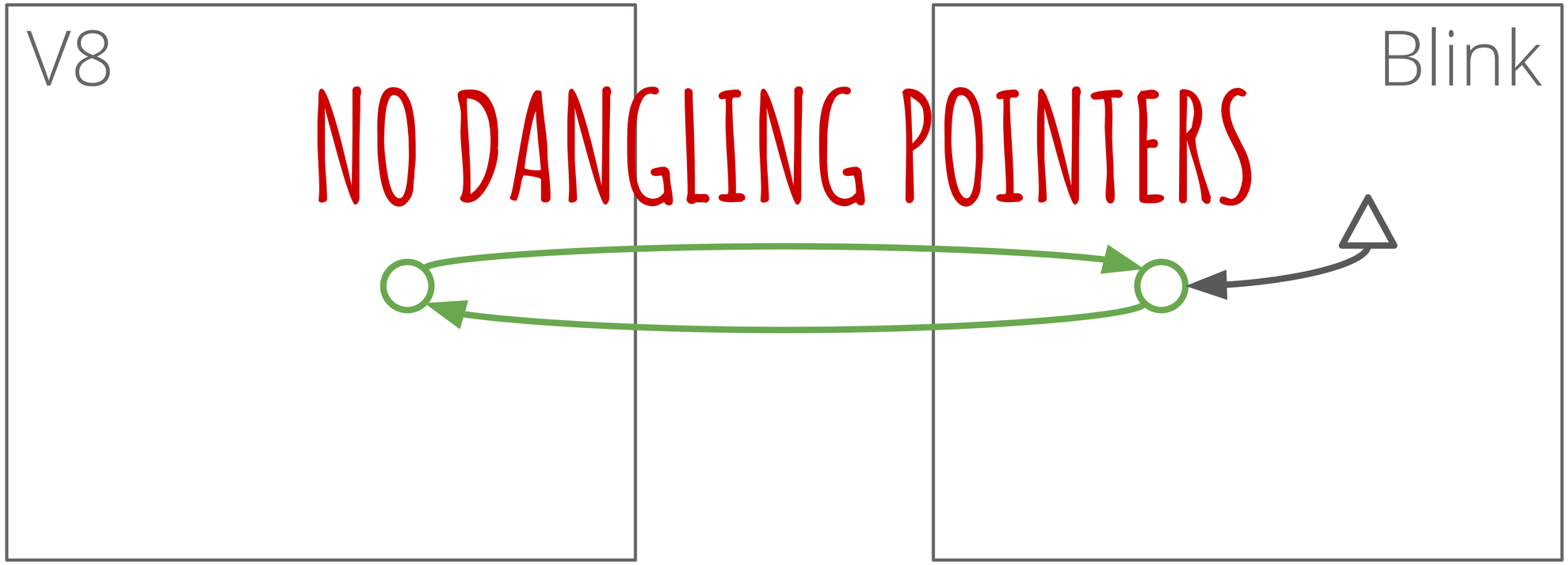
Traced reference



Root reference

Weak reference

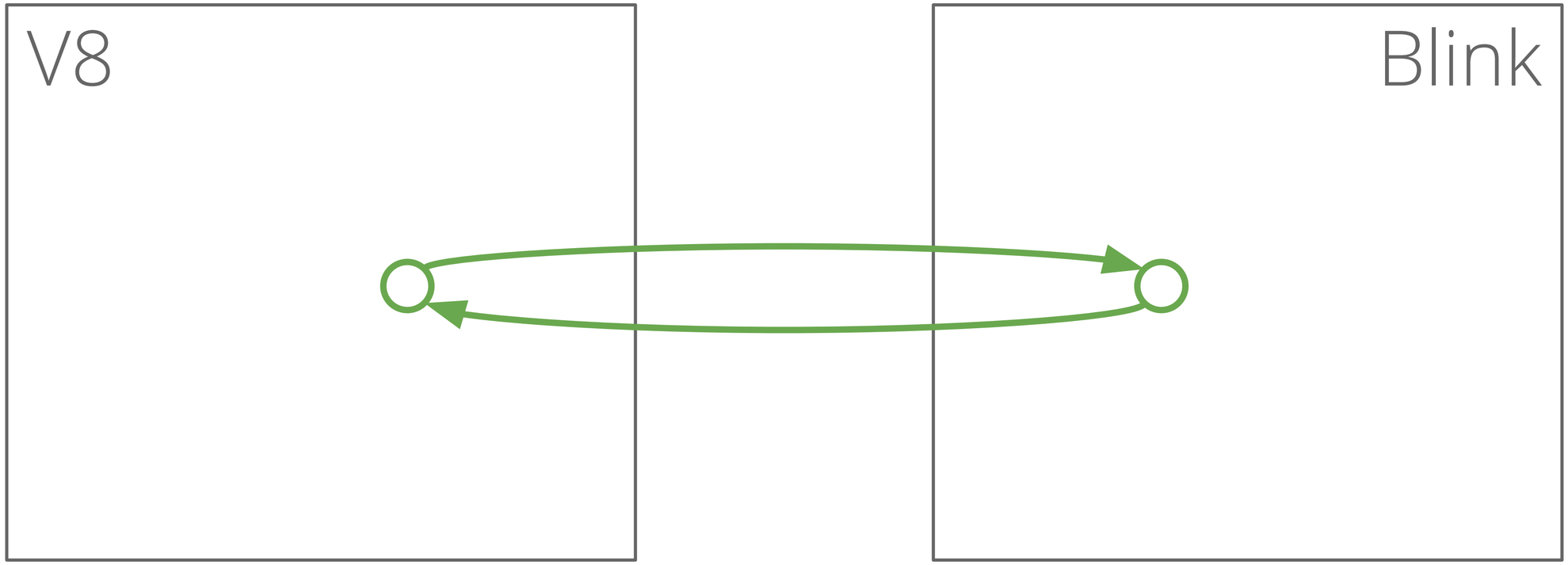
Traced reference



Root reference

Weak reference

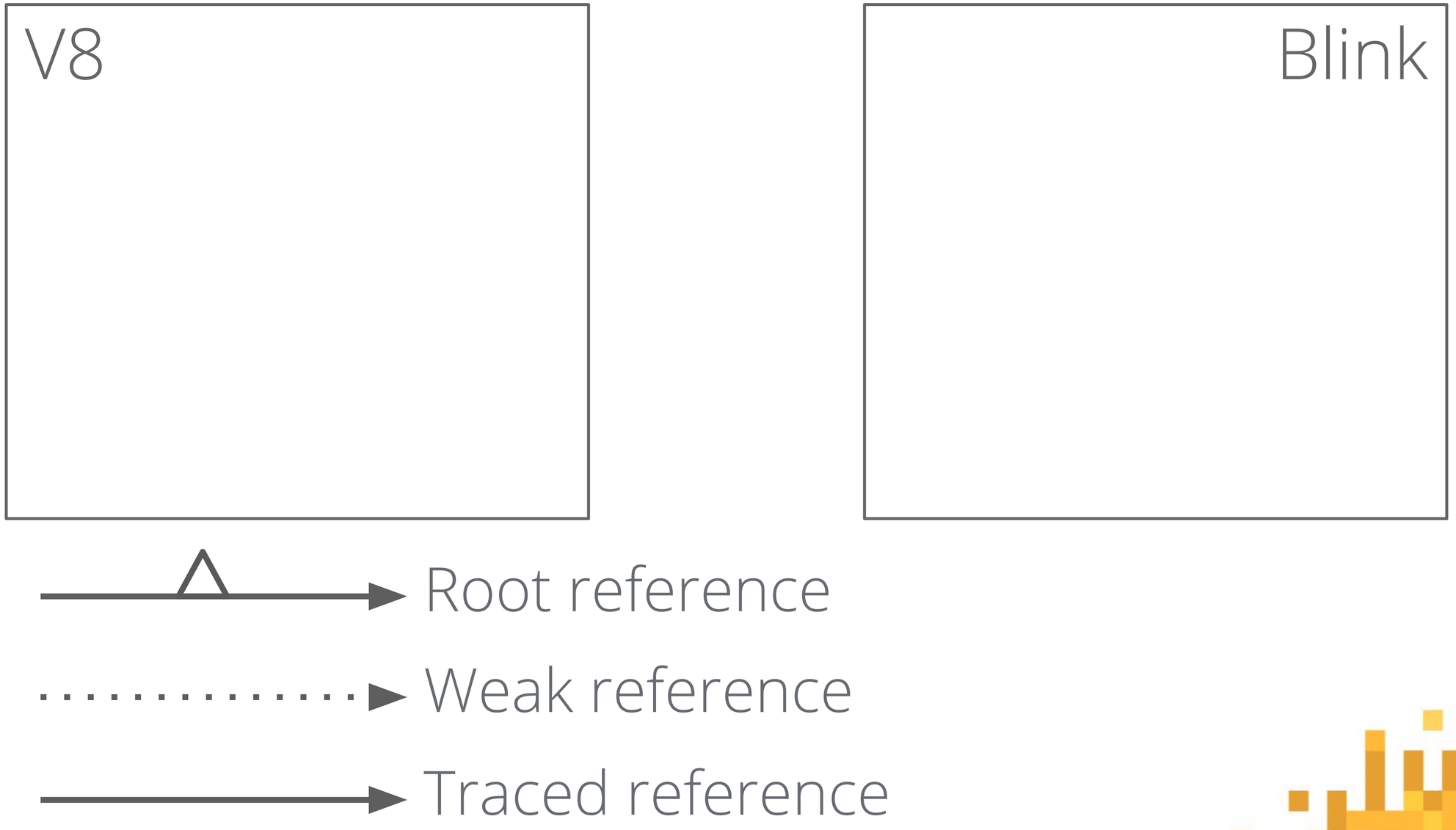
Traced reference

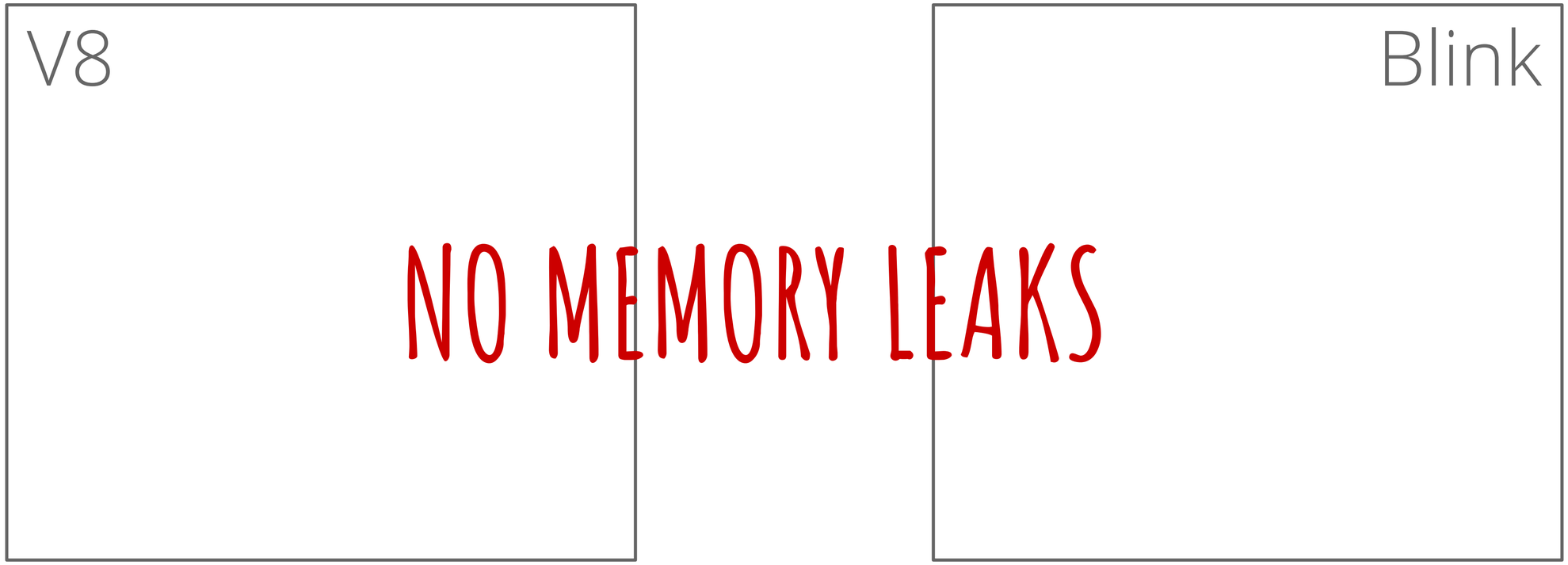


Root reference

Weak reference

Traced reference


V8
Blink
Root reference
Weak reference
Traced reference



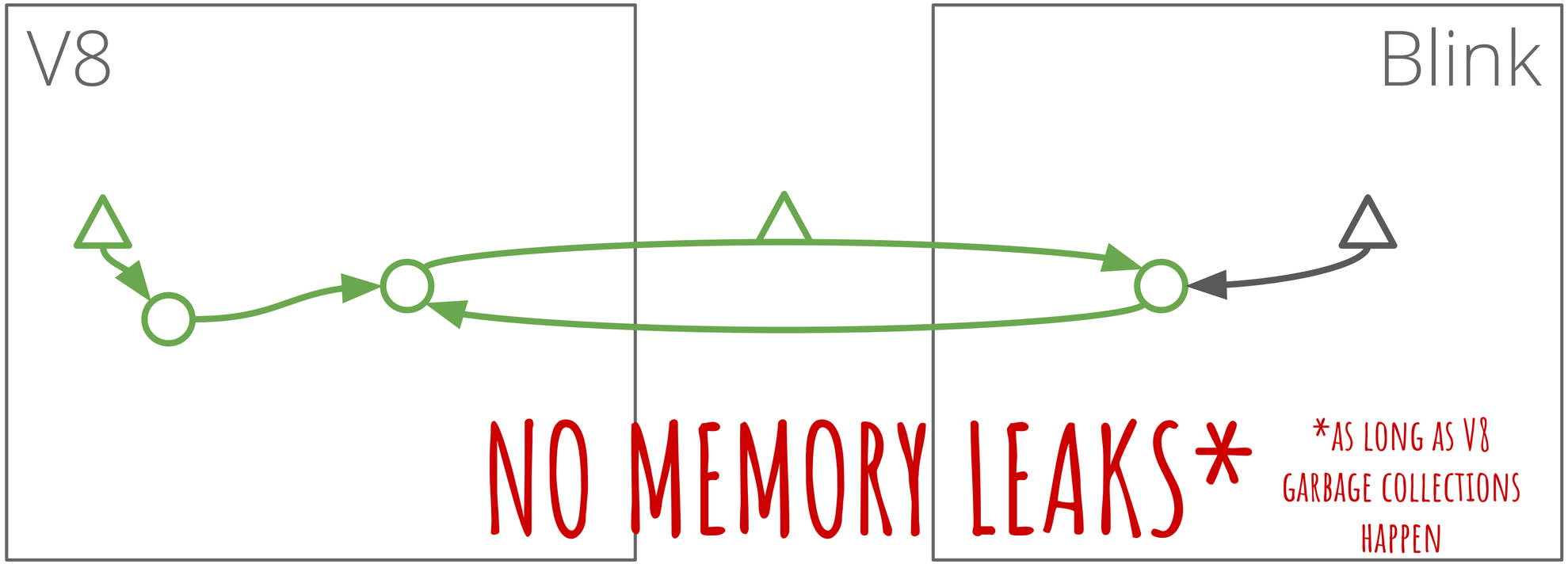
NO MEMORY LEAKS

Root reference

Weak reference

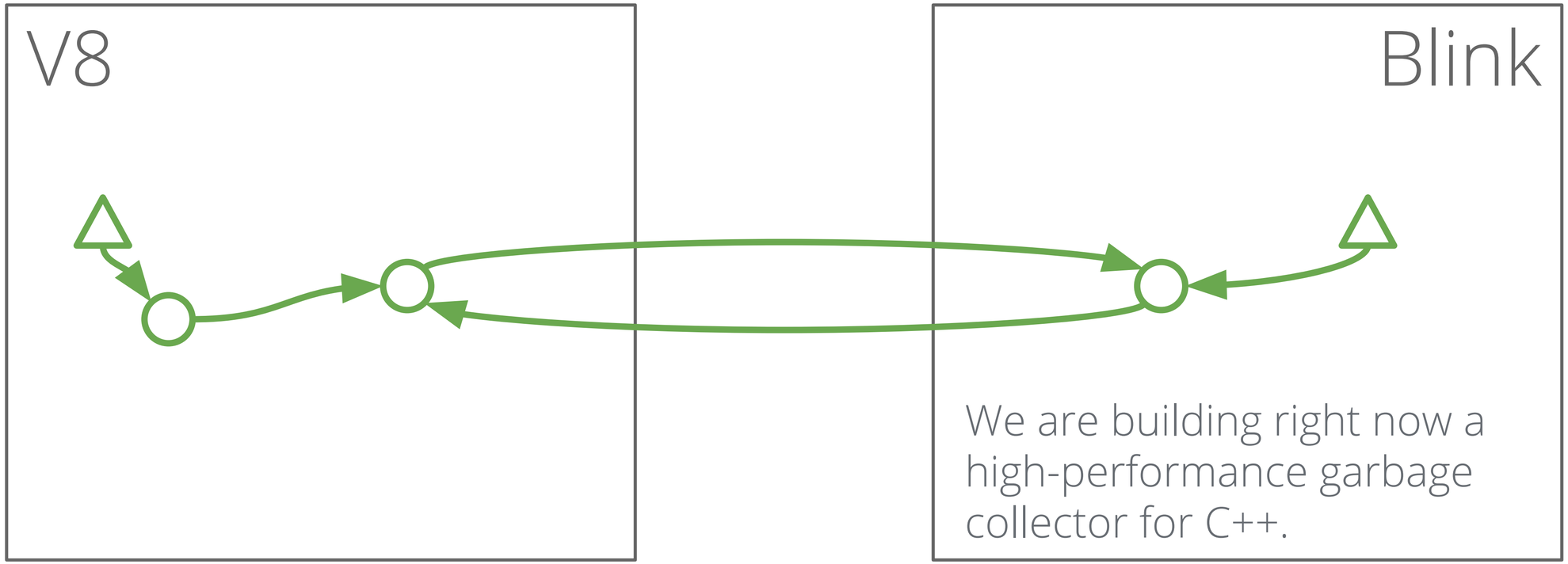
Traced reference

Incremental Wrapper Tracing: From V8 to Blink and Back



Cross-Component Garbage Collection [OOPSLA'18]
Garbage Collection as a Joint Venture [Communications ACM 62(6)]

Today: Unified V8 and Blink Garbage Collection



Thanks!

Hannes Payer
Google|Chrome|V8
https://research.google.com/pubs/HannesPayer.html