Understanding the Java Virtual Machine Memory Management

INTRODUCTION

Kevin Jones

@kevinrjones www.rocksolidknowledge.com

Why GC?

create and forget:
no need to remember
to delete

```
Account acc = new Account();
```

use and forget:
no need to ask
“Should I delete?”

```
Account acc = getAccount();
```

use with confidence:
objects will not
vanish or become corrupt
behind your back

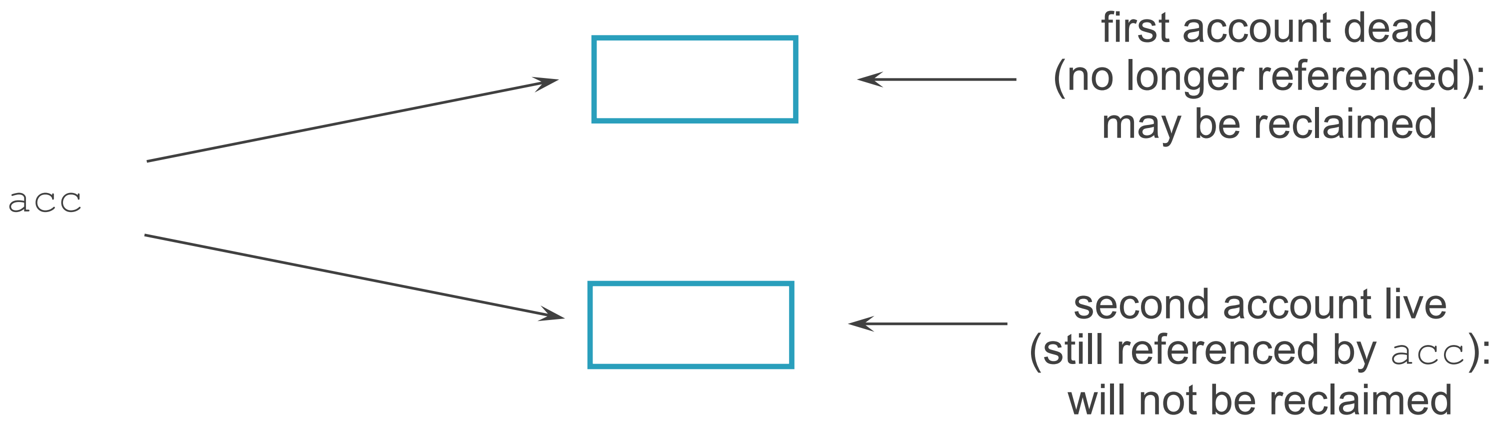
```
acc.increment(amount);
```

The GC Promise

Claim no live objects

- no promises about dead objects

```
Account acc = new Account();
acc = new Account();
```



acc

first account dead
(no longer referenced):
may be reclaimed

second account live
(still referenced by `acc`):
will not be reclaimed

A Note on Versions

Java 8 Still Widely used

- Course runs on Java 11

Things changed in Java 7

- G1 garbage collector introduced

Things changed in Java 9

- CMS garbage collector deprecated
- Finalizers deprecated
- Cleaner introduced

Forms of Garbage Collection

Do Nothing

Reference Counting

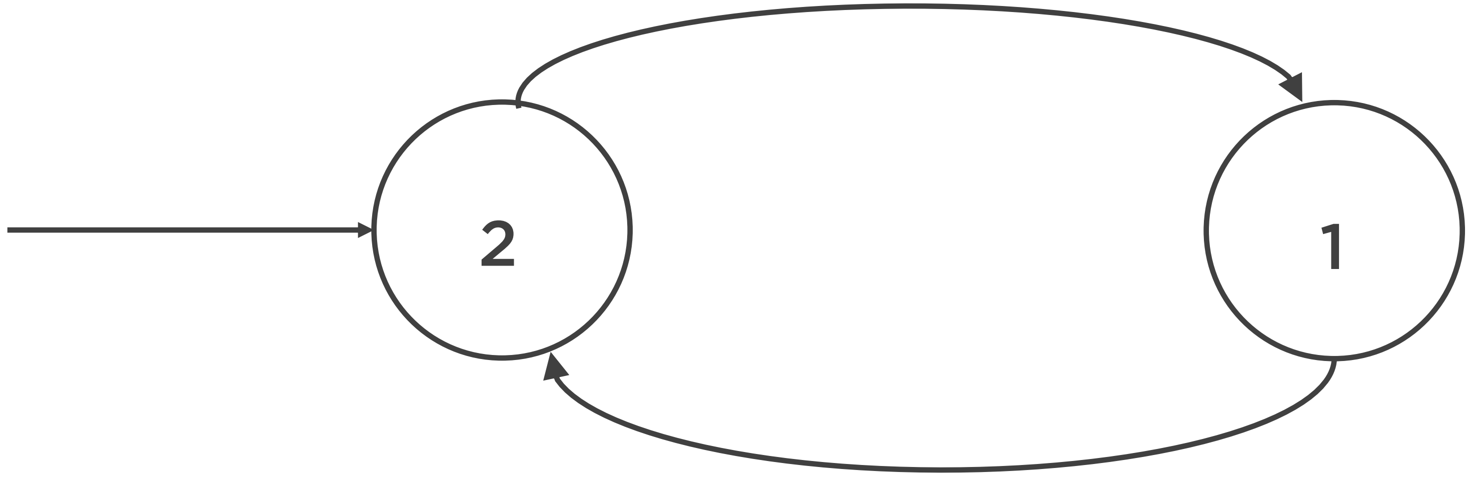
Mark and Sweep

Copying

Generational

Incremental

Reference Counting

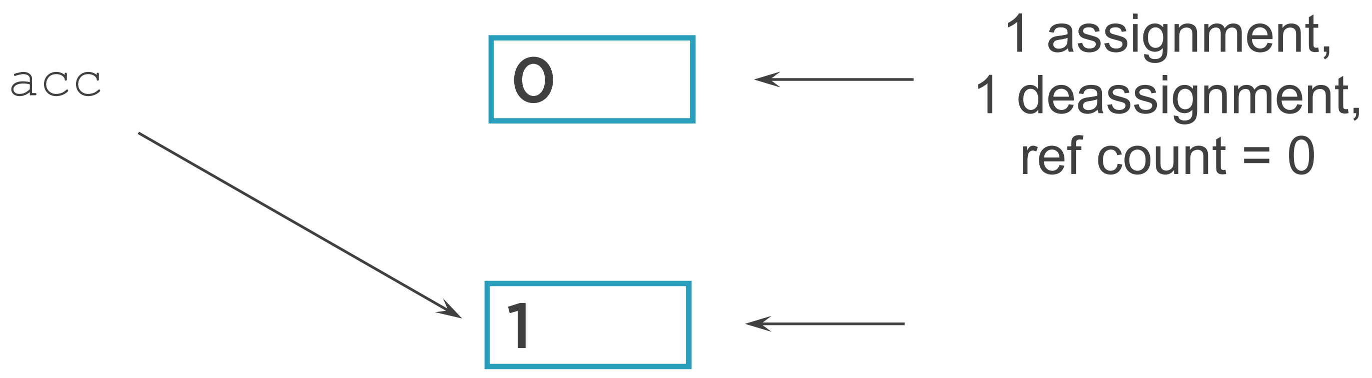


Onus on client to call methods when allocating/freeing memory

- COM for example had AddRef and Release calls for objects
- When count hits zero object can be freed
- Problems with circular references

Reference Counting

```
Account acc = new Account();
acc = new Account();
```



acc

0

1

1 assignment,
1 deassignment,
ref count = 0

Mark and Sweep

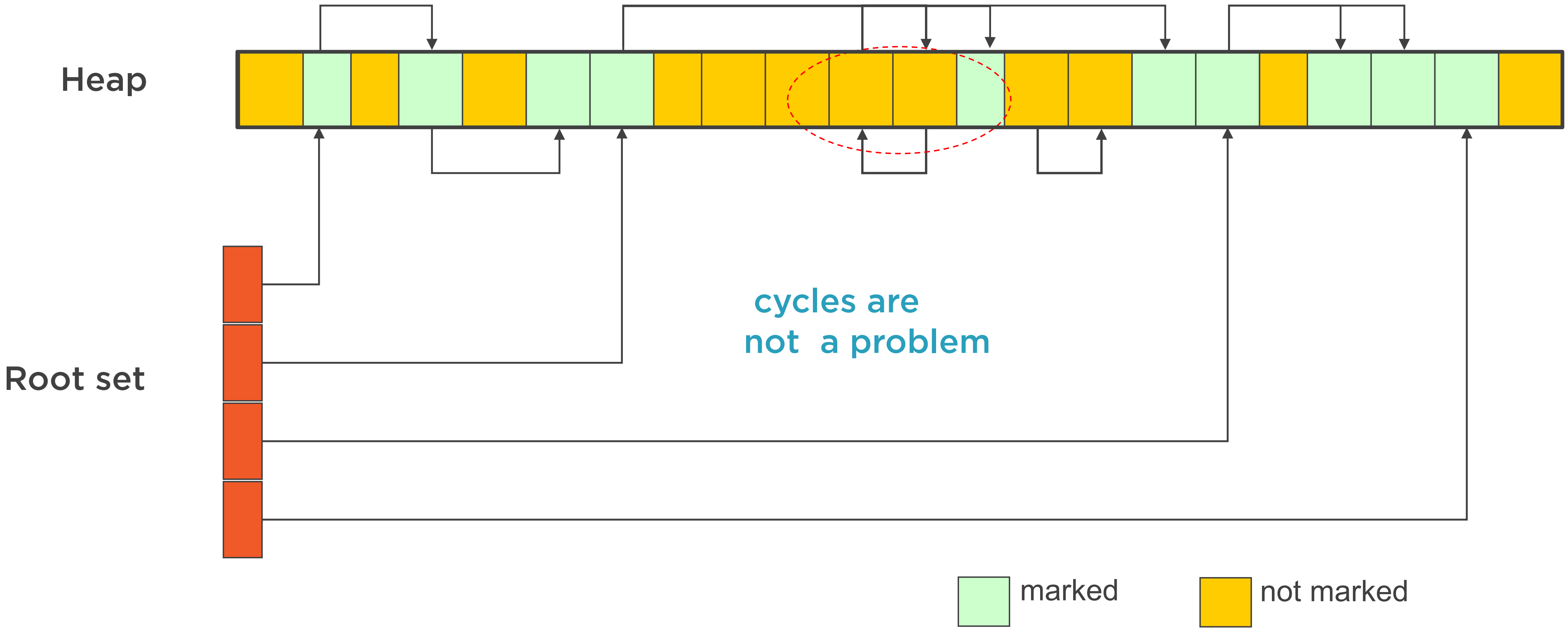
‘mark’ phase that identifies the objects that are still in use

‘sweep’ phase to remove unused objects

‘compact’ phase to compact the memory

Mark Phase

Sweep Phase



Heap

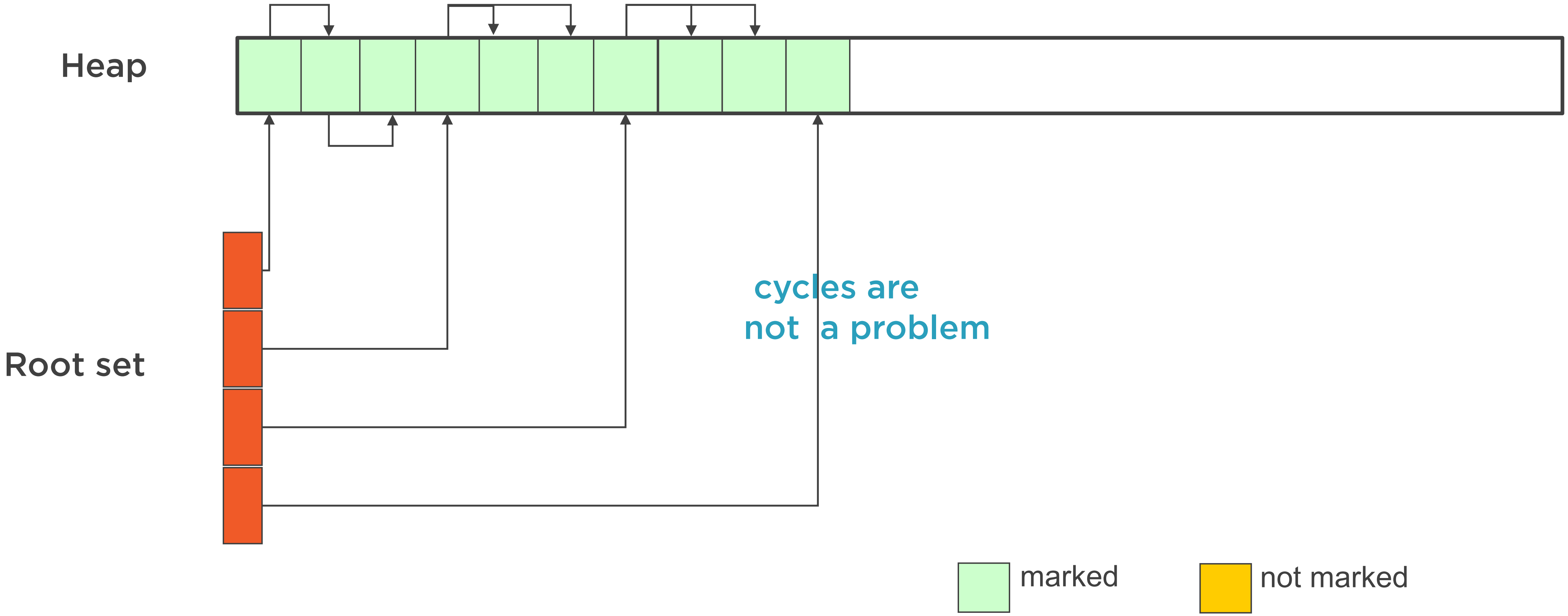
Root set

cycles are not a problem

🟩 marked

🟨 not marked

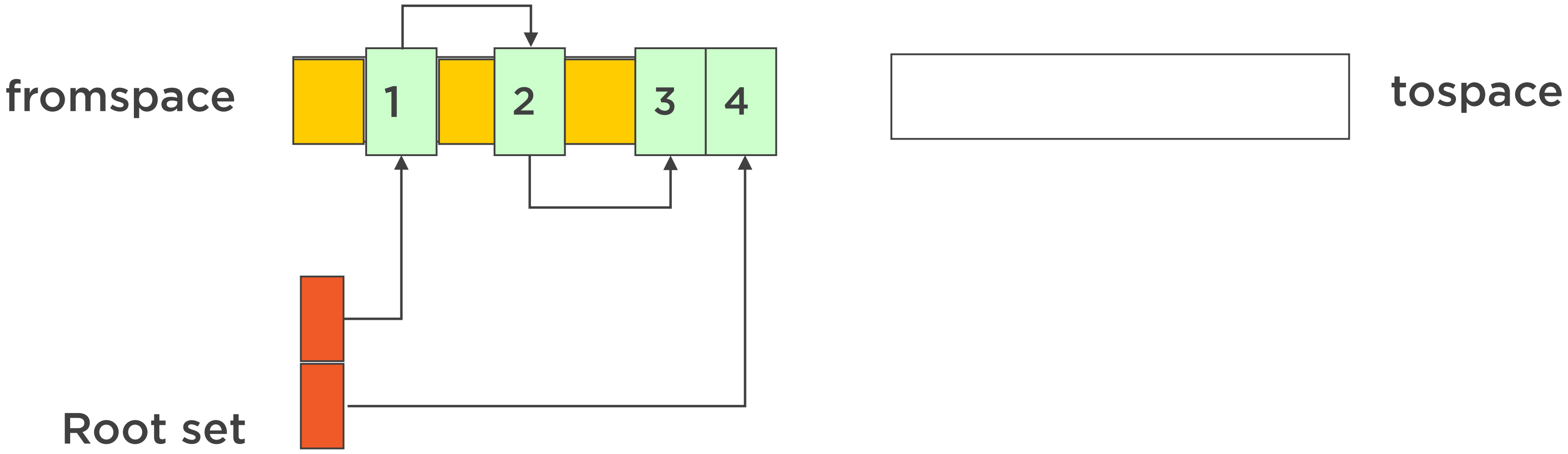
Compact Phase



Copying

Uses different ‘spaces’ to manage memory

Copying Fromspace to Tospace



fromspace

tospace

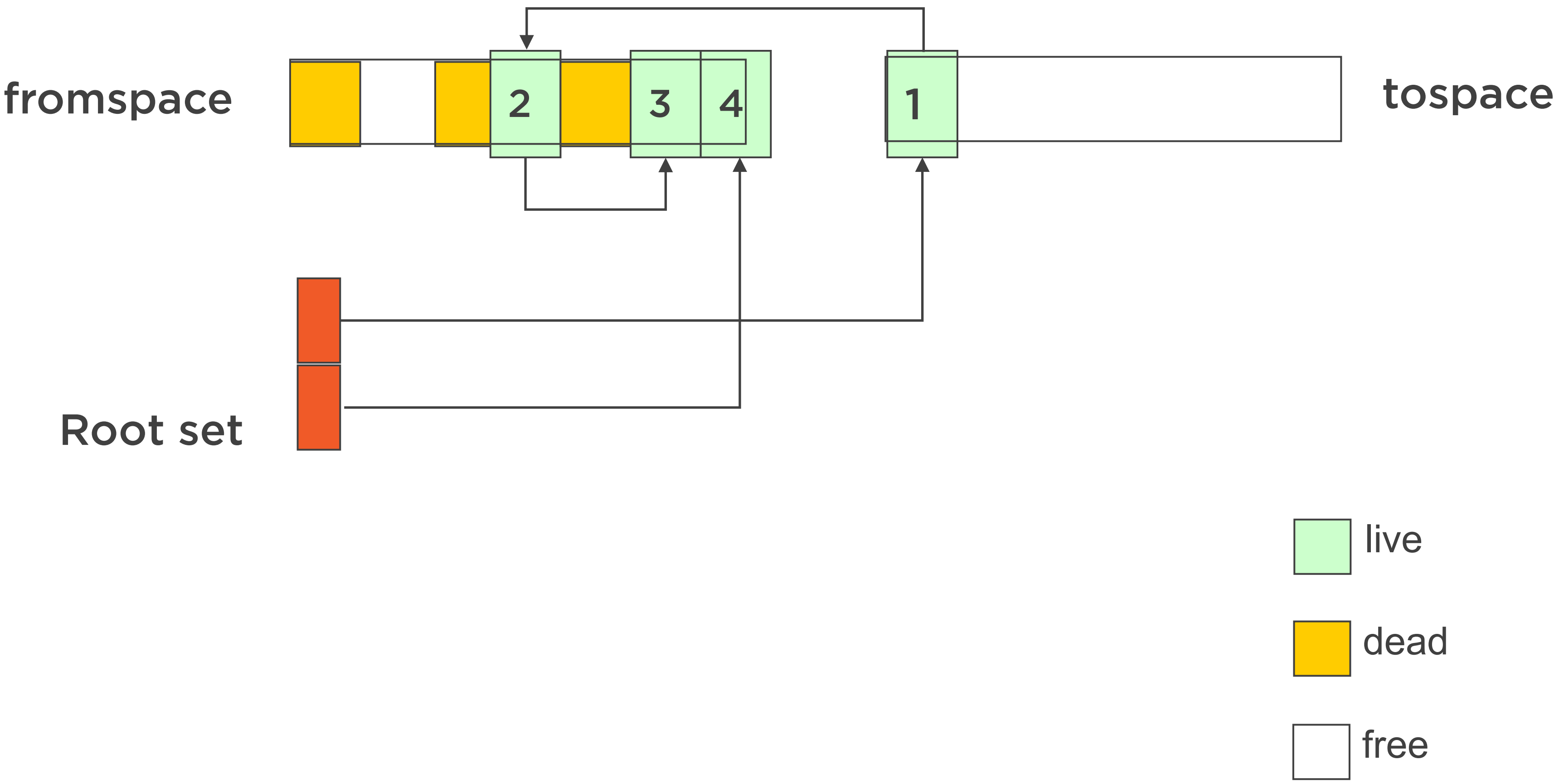
Root set

🟩 live

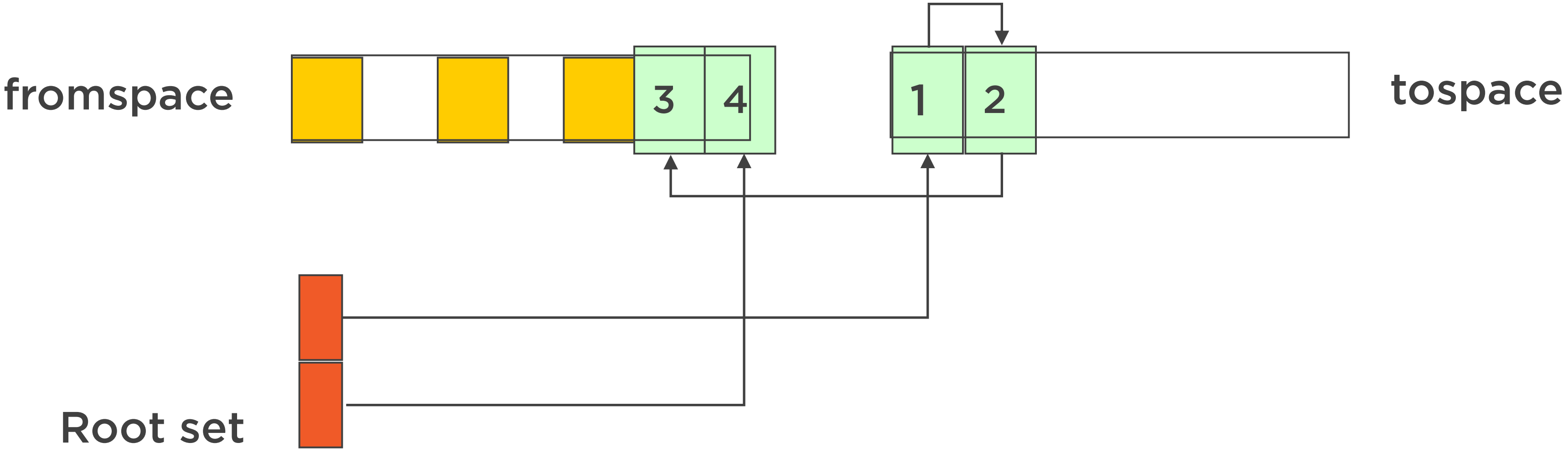
🟨 dead

□ free

Copying Fromspace to Tospace



Copying Fromspace to Tospace



fromspace

tospace

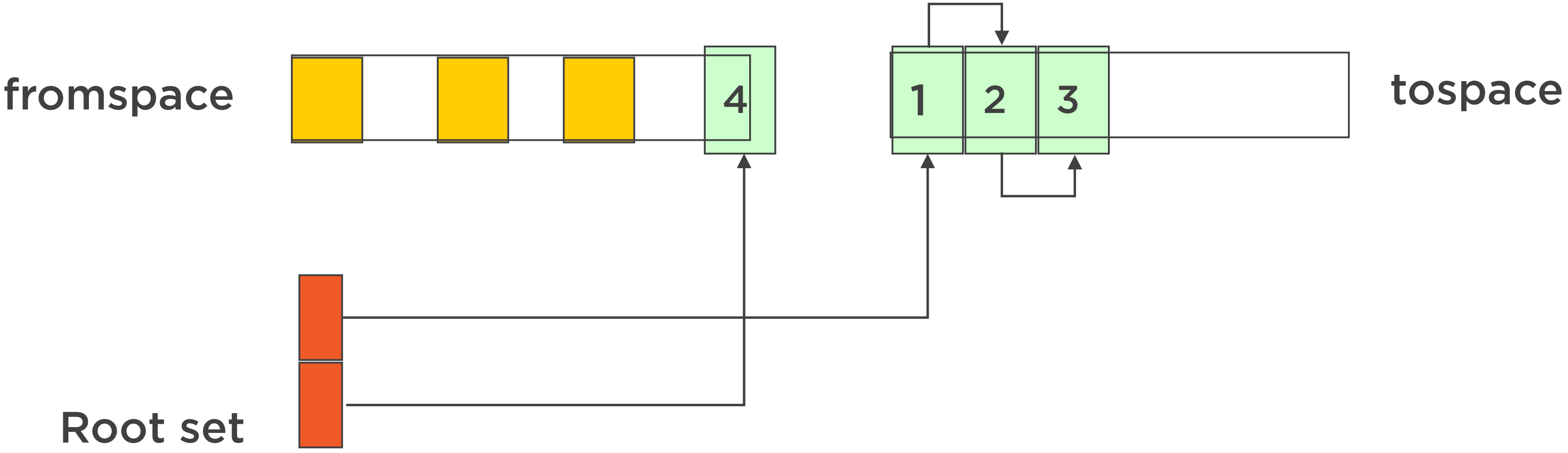
Root set

🟩 live

🟨 dead

□ free

Copying Fromspace to Tospace

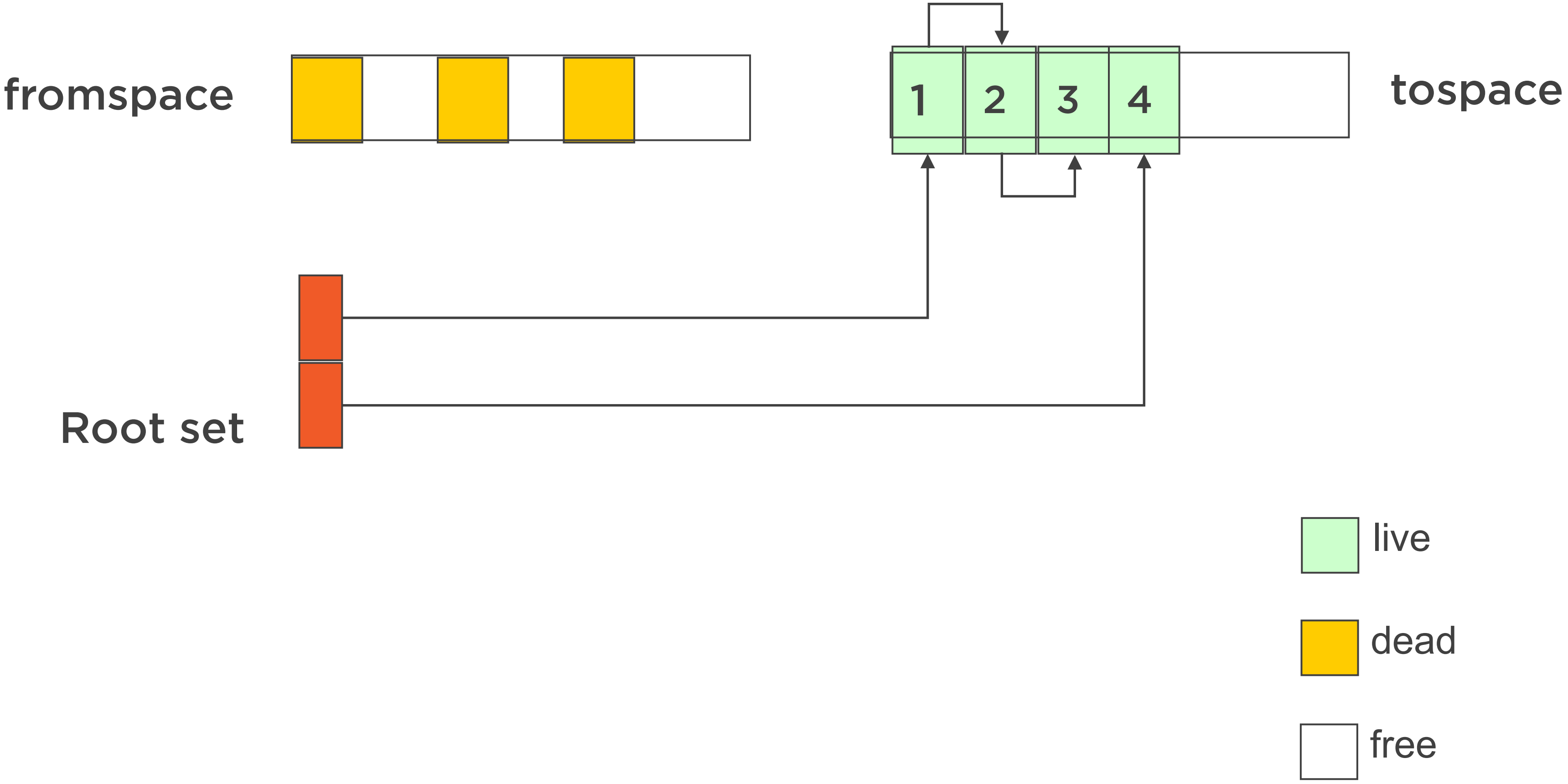


🟩 live

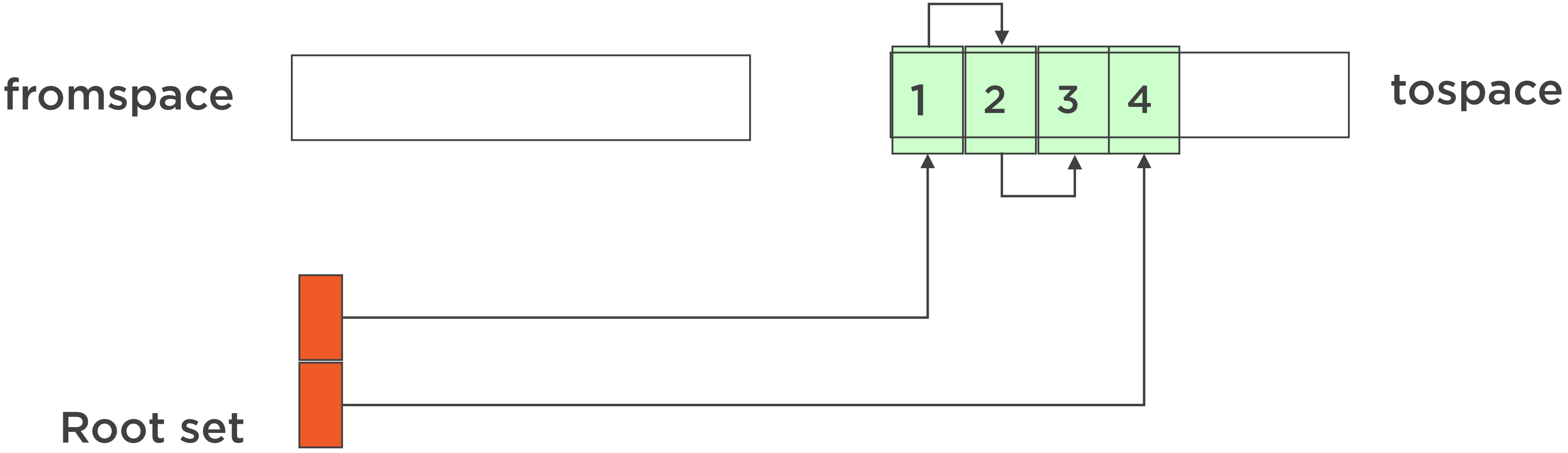
🟨 dead

□ free

Copying Fromspace to Tospace



Copying Fromspace to Tospace



🟩 live

🟨 dead

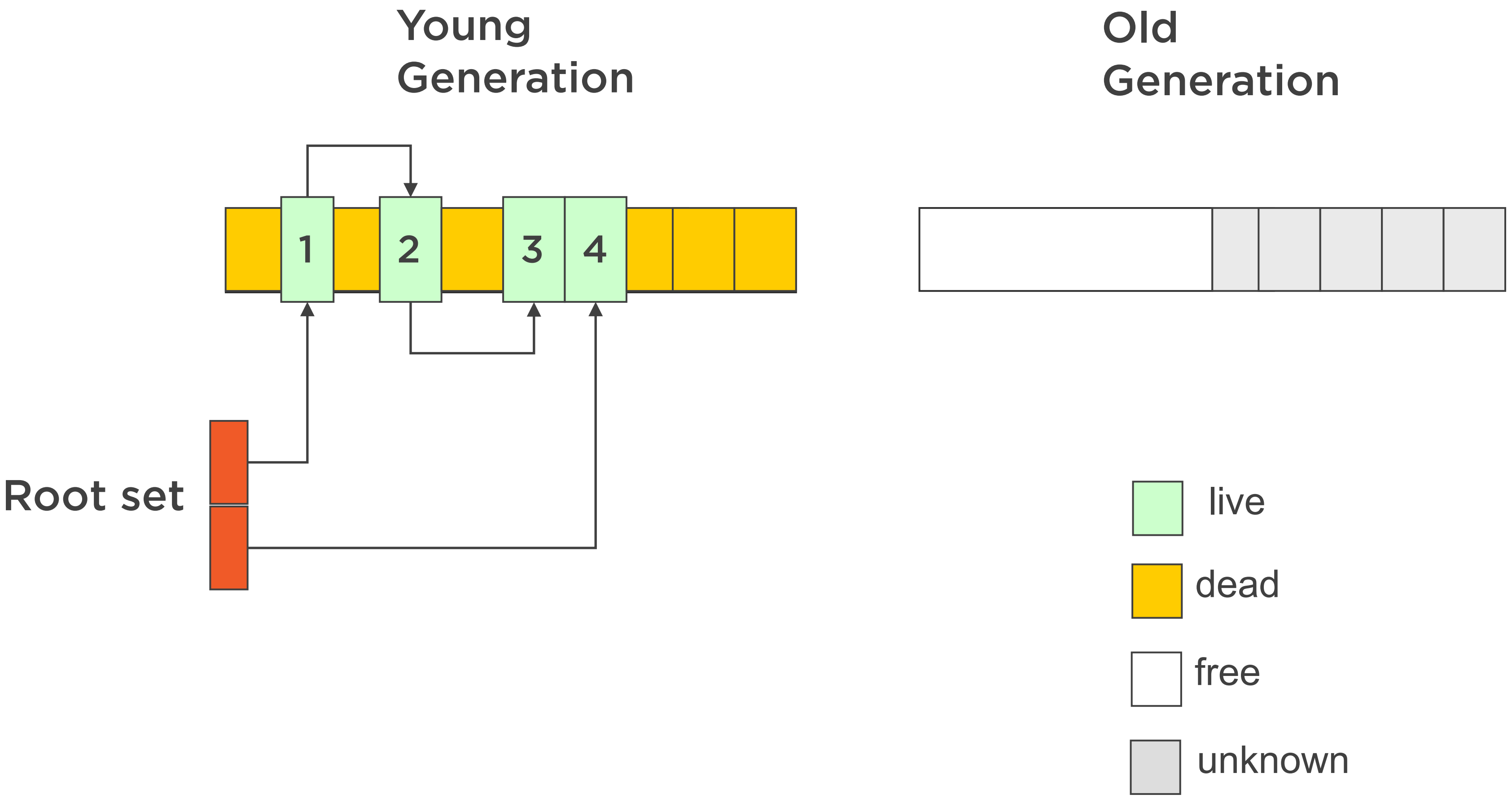
□ free

Generational Collectors

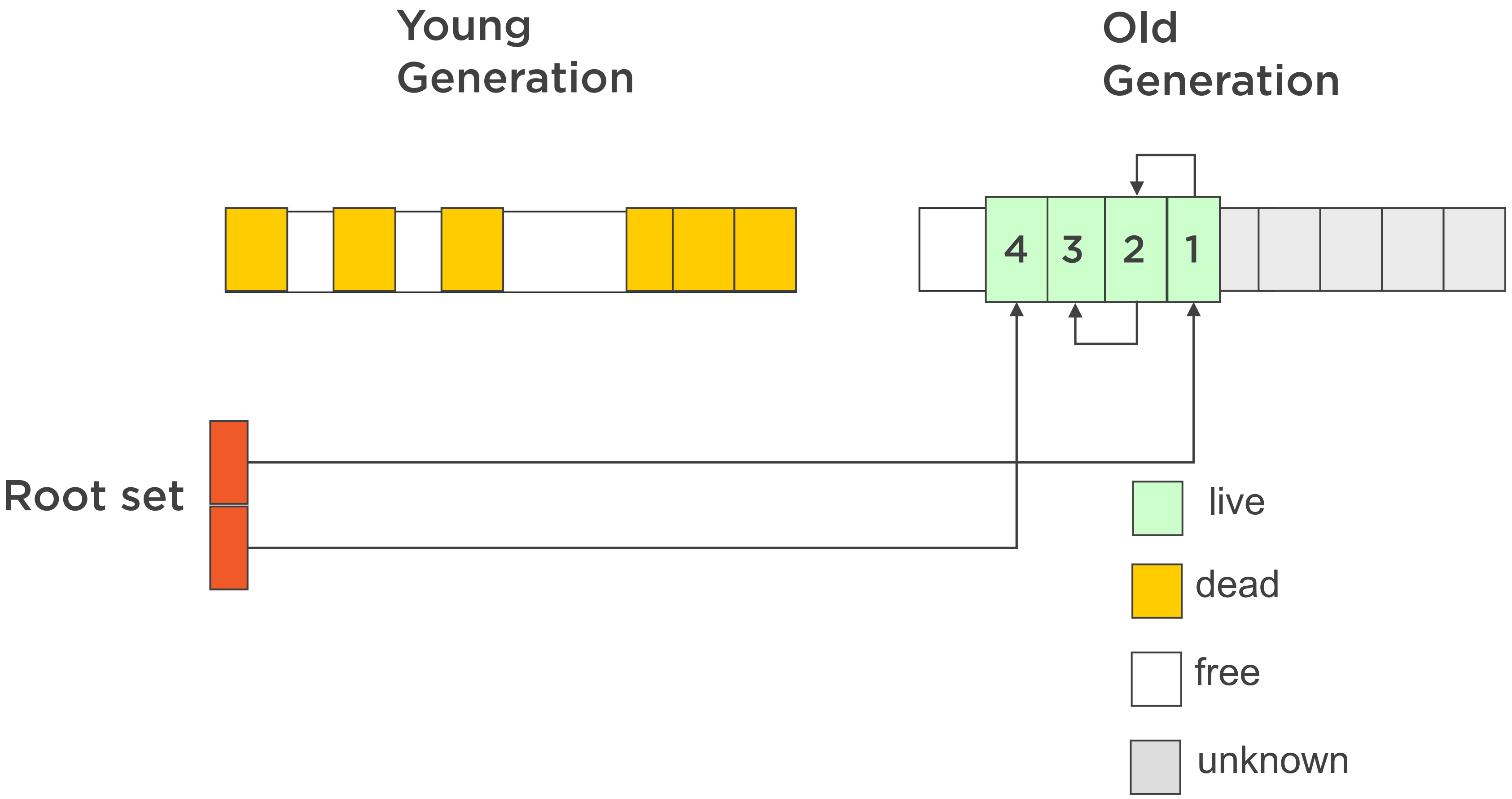
Maintain different generations for memory

- Long living objects ‘promoted’ to a different generation
- For a given definition of ‘long’

Before a Generational Minor Collect



After a Generational Minor Collect



Demo

Showing GC in action