Module: Dynamic Allocator Misuse II

Beyond tcache

Robert Wasinger
Arizona State University

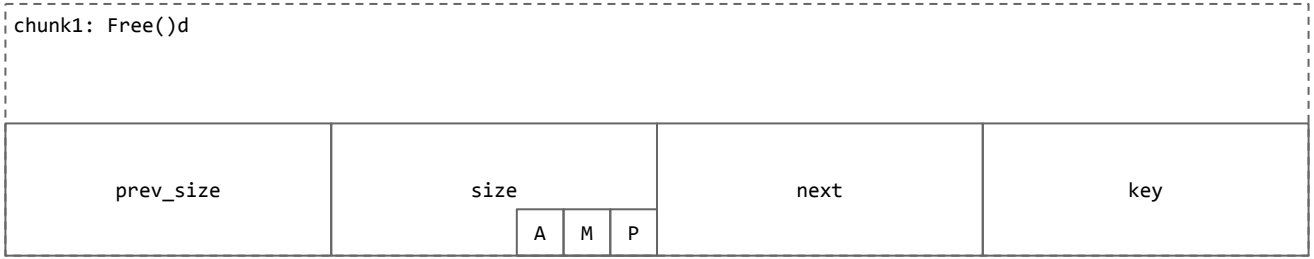
So far... TCACHE

TCACHE

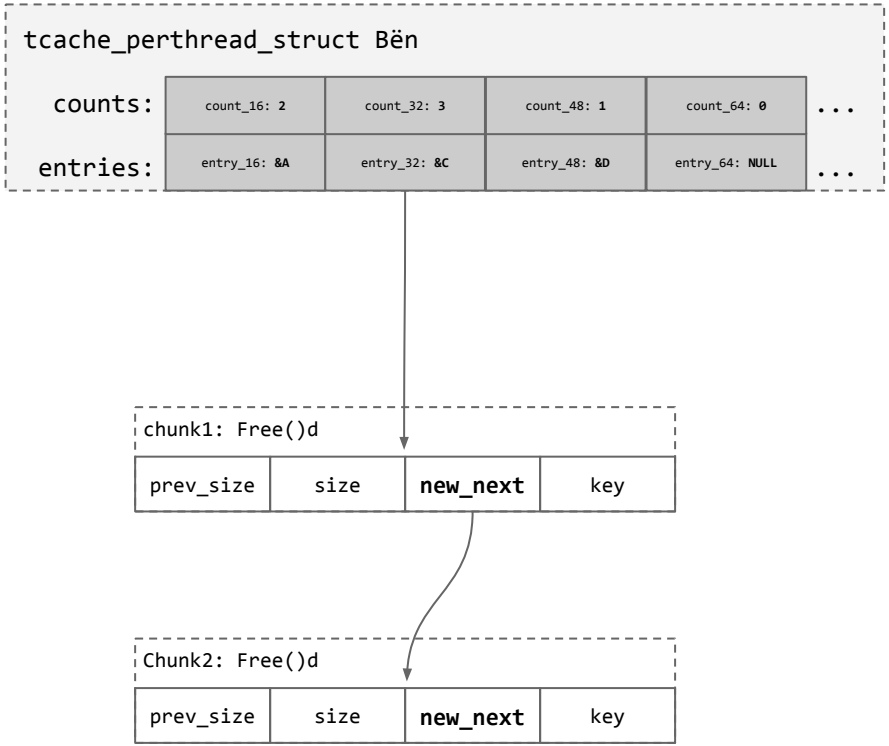
- Bins of constant size up to 1032 bytes
- Caches up to seven freed chunks
- Singly linked list
- Safe-Linking

https://elixir.bootlin.com/glibc/latest/source/malloc/malloc.c#L328

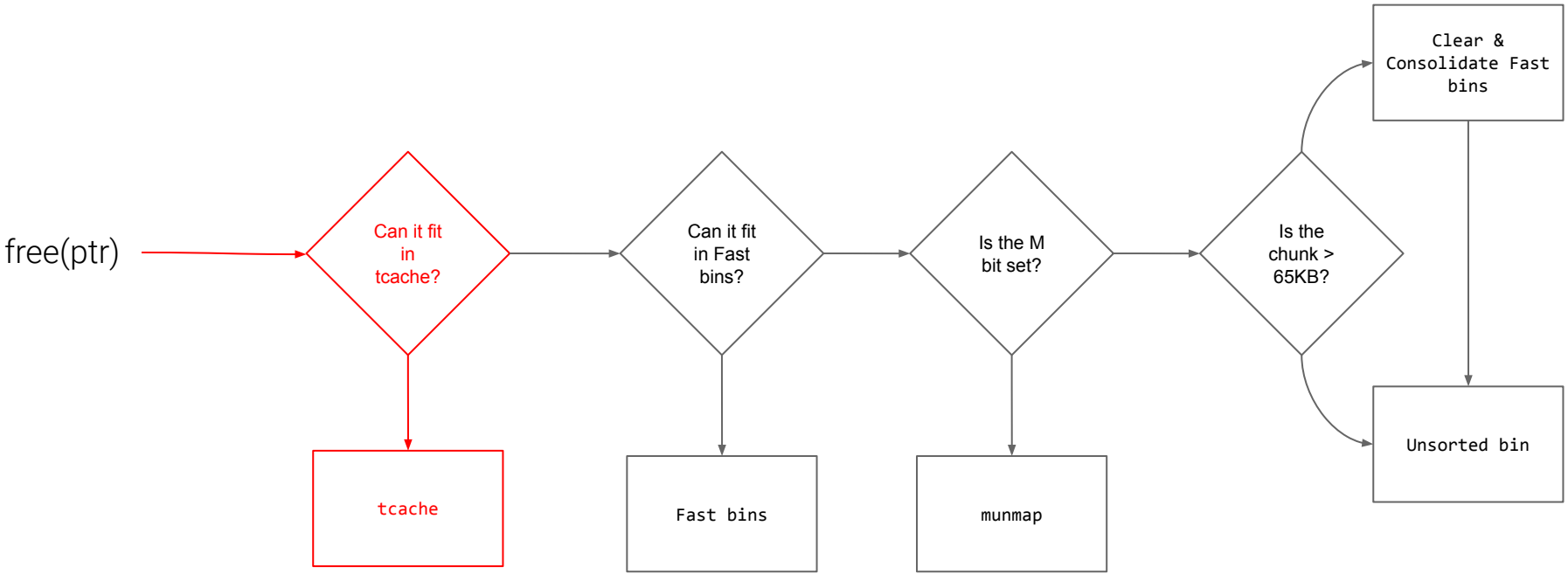
TCACHE Chunk Metadata



TCACHE - Singly Linked List

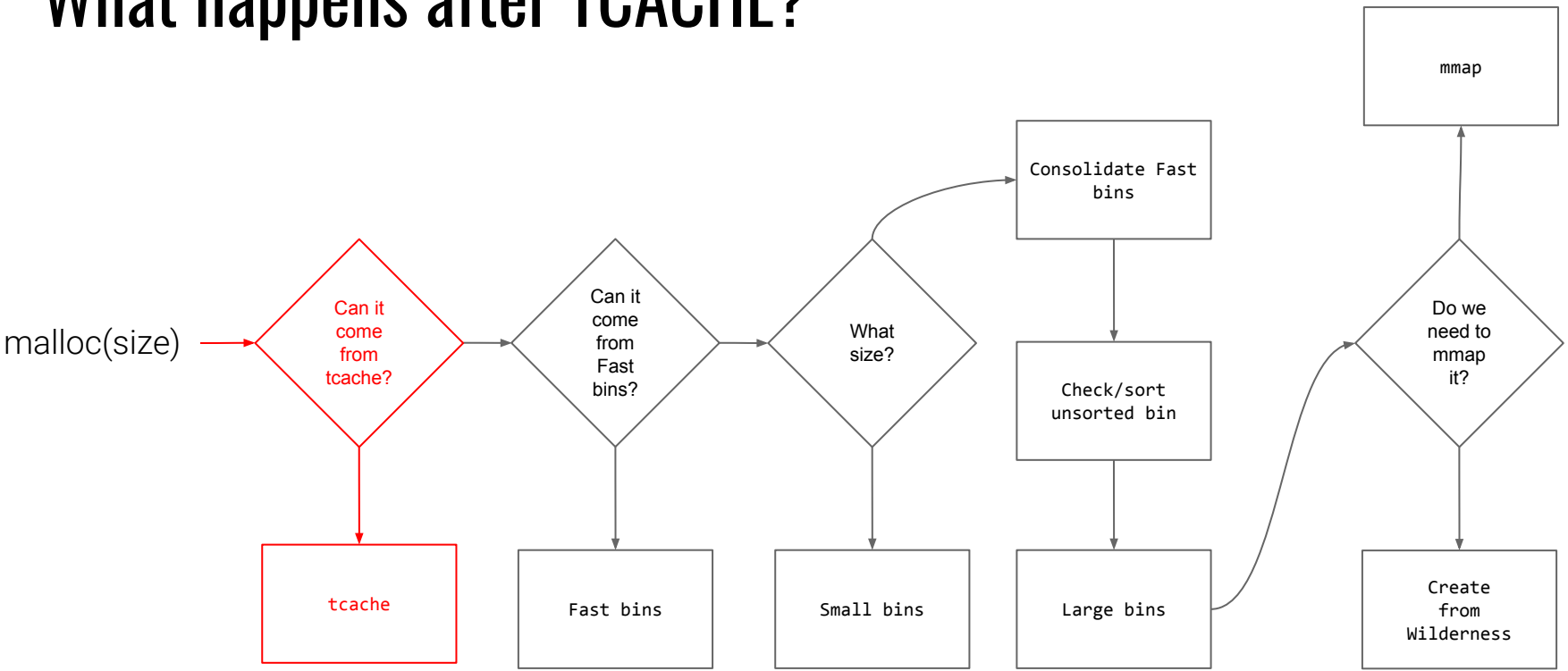


What happens after TCACHE?

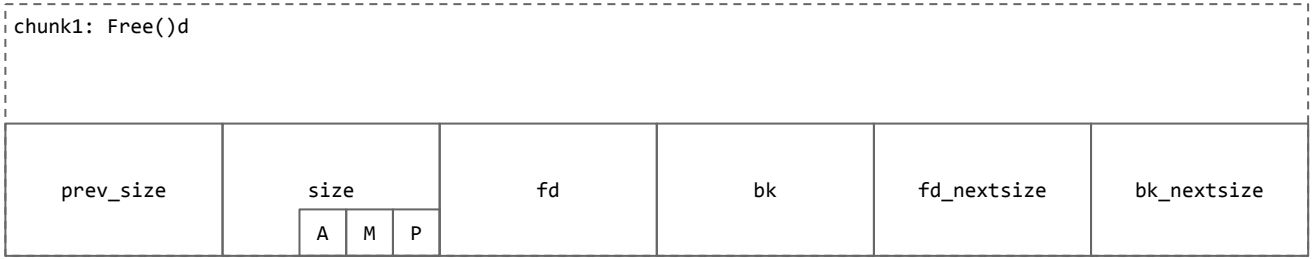


https://elixir.bootlin.com/glibc/latest/source/malloc/malloc.c#L4402

What happens after TCACHE?

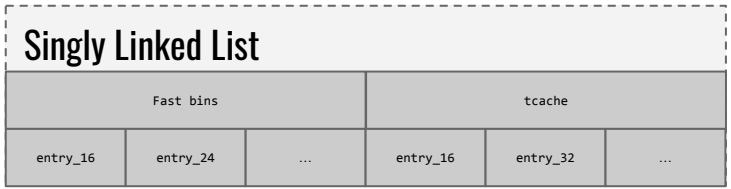


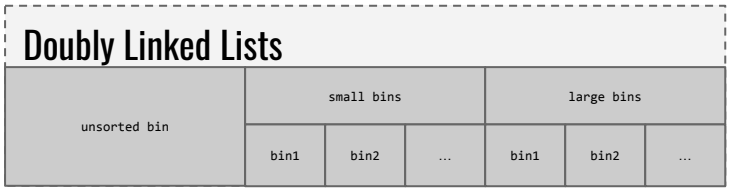
More metadata



https://elixir.bootlin.com/glibc/latest/source/malloc/malloc.c#L1130

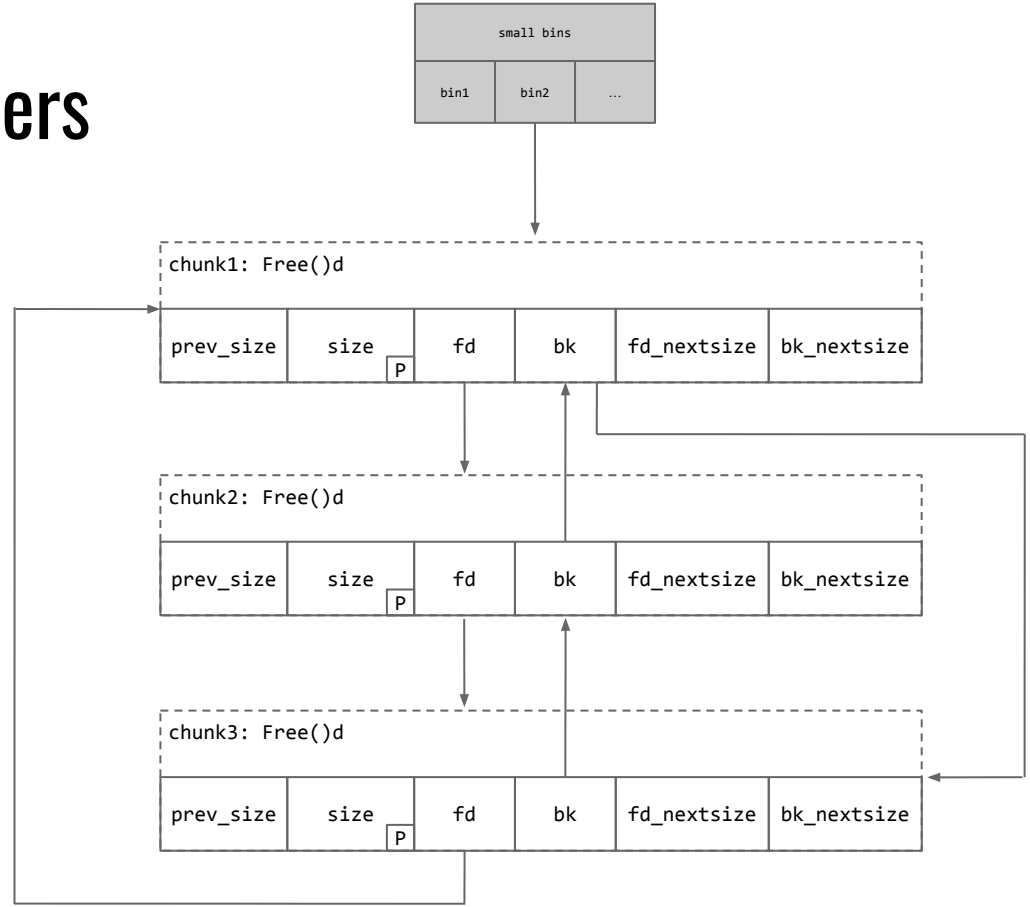
More lists





https://elixir.bootlin.com/glibc/latest/source/malloc/malloc.c#L1130

More pointers



https://elixir.bootlin.com/glibc/latest/source/malloc/malloc.c#L1130

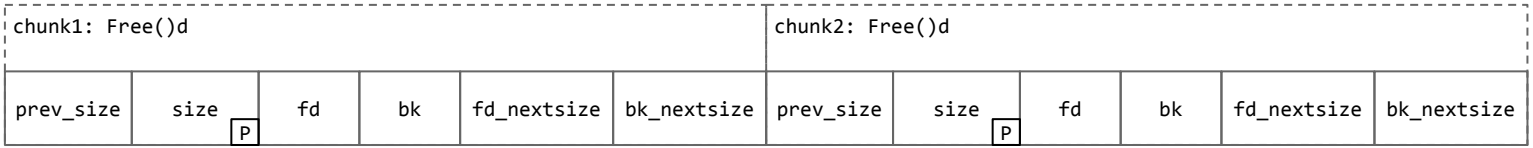
Consolidation

- Consolidation is why a doubly linked list is necessary
- Combines two neighboring chunks
- This can occur:
  - When a chunk is freed
  - When a chunk is malloc'd
- Consolidation requires the removal of an entry after merging

https://elixir.bootlin.com/glibc/latest/source/malloc/malloc.c#L4700

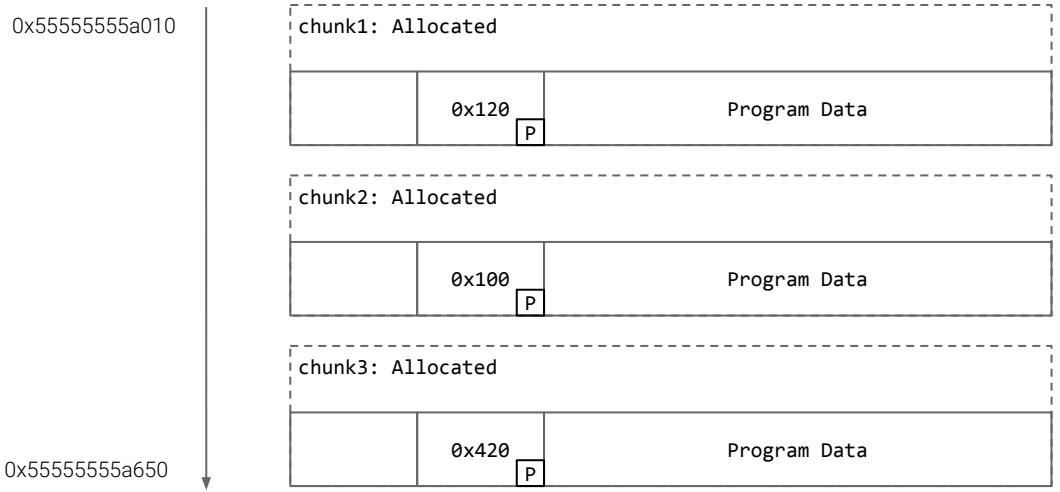
Consolidation

- Consolidation occurs forward and backward!
- The P bit must be cleared for a chunk to consolidate



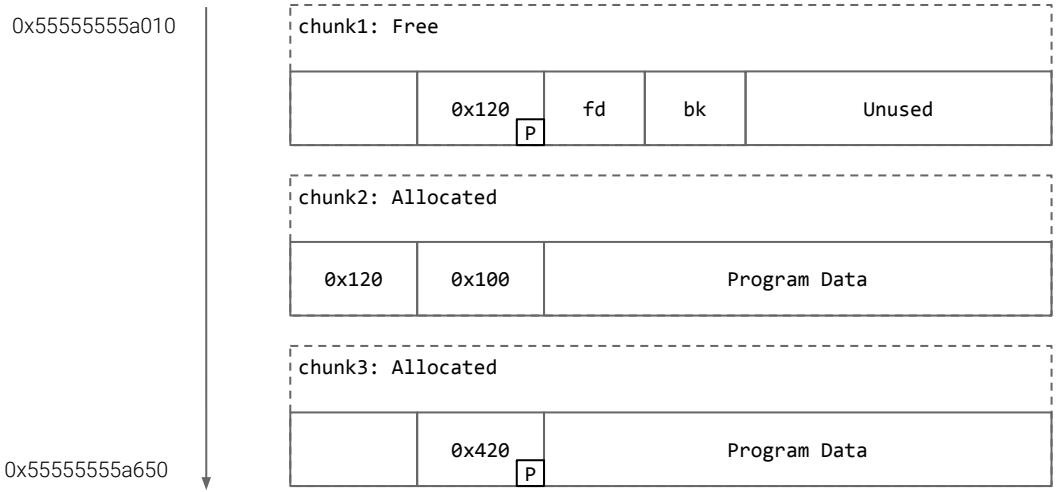
Consolidation - In Memory

Initial state



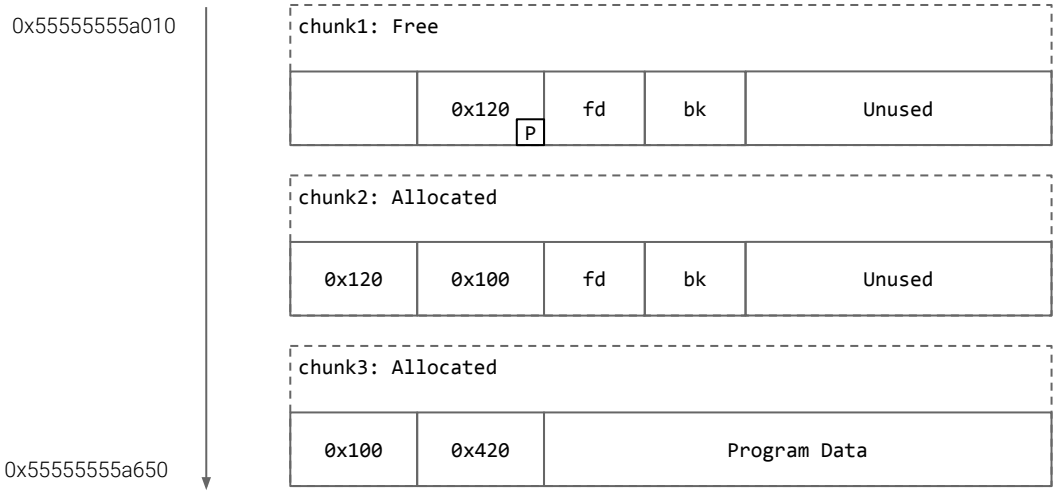
Consolidation - In Memory

`free(0x55555555a028)`



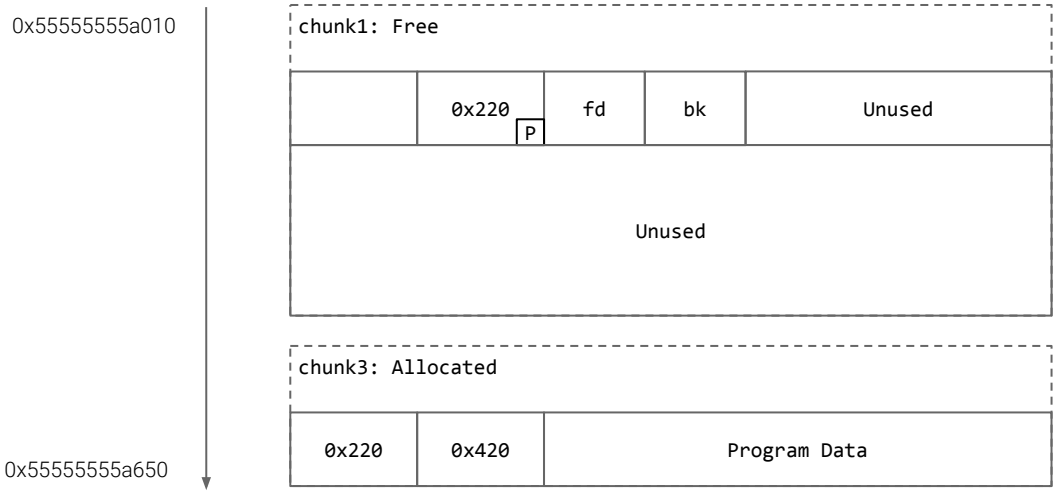
Consolidation - In Memory

`free(0x55555555a028)`

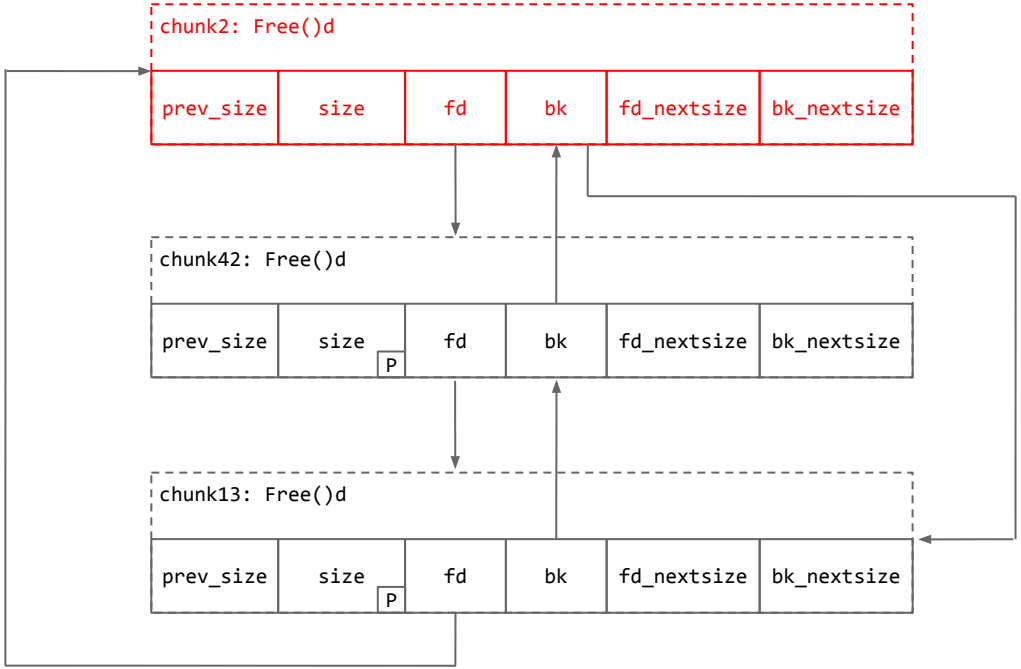


Consolidation - In Memory

`free(0x55555555a148)`

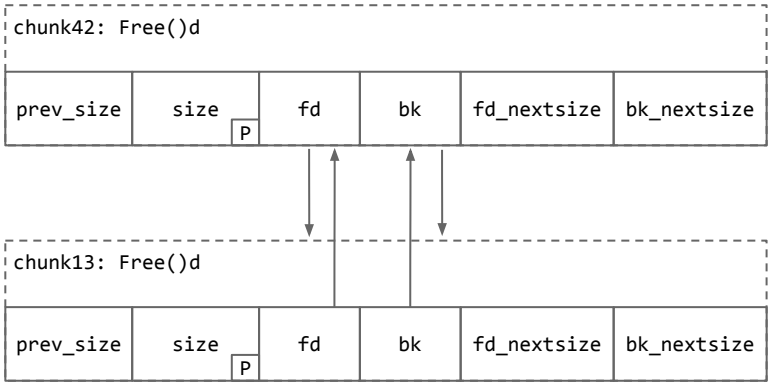


Consolidation - Unlinking



https://elixir.bootlin.com/glibc/latest/source/malloc/malloc.c#L1602

Consolidation - Unlinking



https://elixir.bootlin.com/glibc/latest/source/malloc/malloc.c#L1602

Consolidation - Unlinking verification

```
/* Take a chunk off a bin list.  */

static void unlink_chunk (mstate av, mchunkptr p) {

 if (chunksize (p) != prev_size (next_chunk (p)))

    malloc_printerr ("corrupted size vs. prev_size");

  mchunkptr fd = p->fd;

  mchunkptr bk = p->bk;

  if (__builtin_expect (fd->bk != p || bk->fd != p, 0))

    malloc_printerr ("corrupted double-linked list");

  fd->bk = bk;

  bk->fd = fd;
```

https://elixir.bootlin.com/glibc/latest/source/malloc/malloc.c#L1602