

Appendix: Remaining Eighteen No Durability Guarantee Bugs in Memcached

```
1 void do_slabs_free(...) {
2     ...
3     it->it_flags = ITEM_SLABBED; // First bug
4     if (pslab_contains((char *)it)) {
5         it->it_flags |= ITEM_PSLAB; // Second bug
6     }
7     it->slabs_clsid = 0; // Third bug
8     it->prev = 0; // Fourth bug
9     it->next = p->slots; // Fifth bug
10    if (it->next) {
11        it->next->prev = it; // Six bug
12    }
13    ...
14 }
15
16 static void *do_slabs_alloc(...) {
17     ...
18     it = (item *)p->slots;
19     p->slots = it->next;
20     if (it->next) {
21         it->next->prev = 0; // Seventh bug
22     }
23     ...
24 }
```

Figure 1: No durability guarantee bugs in slab.c.

```
1 int lru_pull_tail(...) {
2     ...
3     if ((search->it_flags & ITEM_ACTIVE) != 0) {
4         search->it_flags &= ~ITEM_ACTIVE; // Eighth bug
5     }
6     ...
7     ...
8     if (move_to_lru) {
9         it->slabs_clsid = ITEM_clsid(it);
10        it->slabs_clsid |= move_to_lru; // Ninth bug
11        item_link_q(it);
12    }
13    ...
14 }
15
16 void do_item_remove(item *it) {
17     ...
18     if (refcount_decr(it) == 0) {
19         ...
20         if (refcount_decr(it) == 0) { // Tenth bug
21             item_free(it);
22         }
23         ...
24 }
```

Figure 2: No durability guarantee bugs in items.c.

```
1 static void do_item_unlink_q(...) {
2     ...
3     if (it->next) {
4         it->next->prev = it->prev; // Eleventh bug
5     }
6     if (it->prev) {
7         it->prev->next = it->next; // Twelfth bug
8     }
9     ...
10 }
11
12 item *do_item_crawl_q(...) {
13     ...
14     if (it->next) {
15         assert(it->prev->next == it);
16         it->prev->next = it->next; // Thirteenth bug
17         it->next->prev = it->prev; // Fourteenth bug
18     }
19     ...
20 }
21
22 do_item_unlinktail_q(...) {
23     ...
24     if (it->next) {
25         it->next->prev = it->prev; // Fifteenth bug
26     }
27     if (it->prev) {
28         it->prev->next = it->next; // Sixteenth bug
29     }
30     return;
31 }
32
33 void do_item_update(...) {
34     ...
35     if (settings.lru_segmented) {
36         ...
37         it->slabs_clsid |= WARM_LRU; // Seventeenth bug
38         ...
39         else if (it->time < current_time -
40                 ITEM_UPDATE_INTERVAL) {
41             it->time = current_time; // Eighteenth bug
42         }
43         ...
44 }
```

Figure 3: No durability guarantee bugs in items.c.