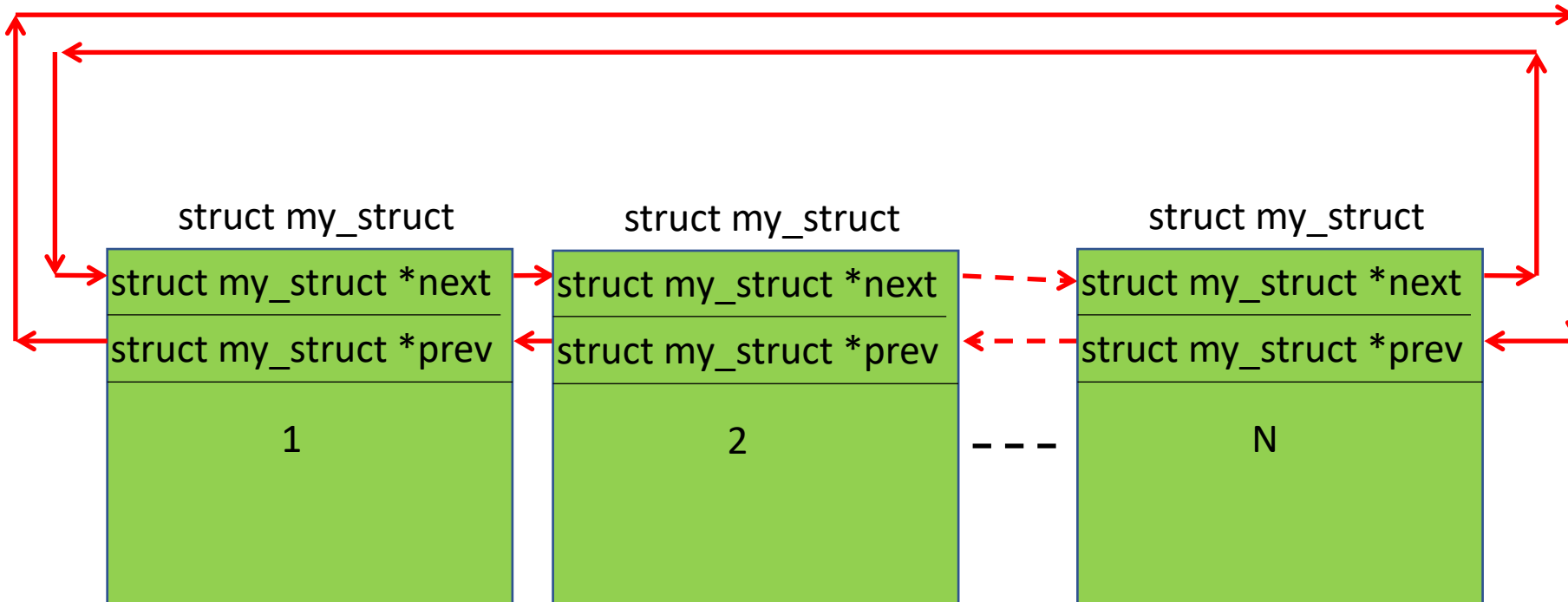


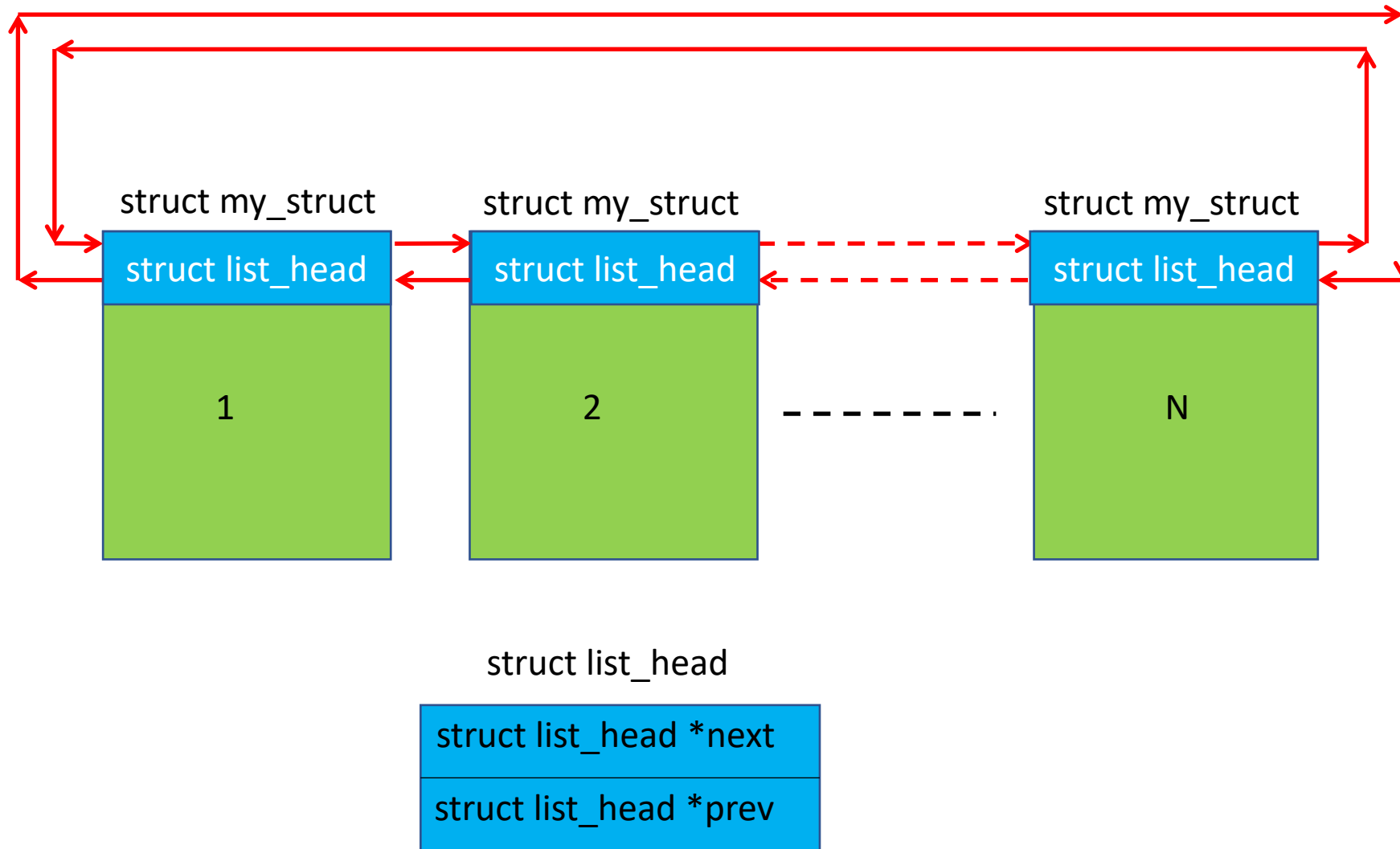
Linux List-Head

By Jitesh Verma

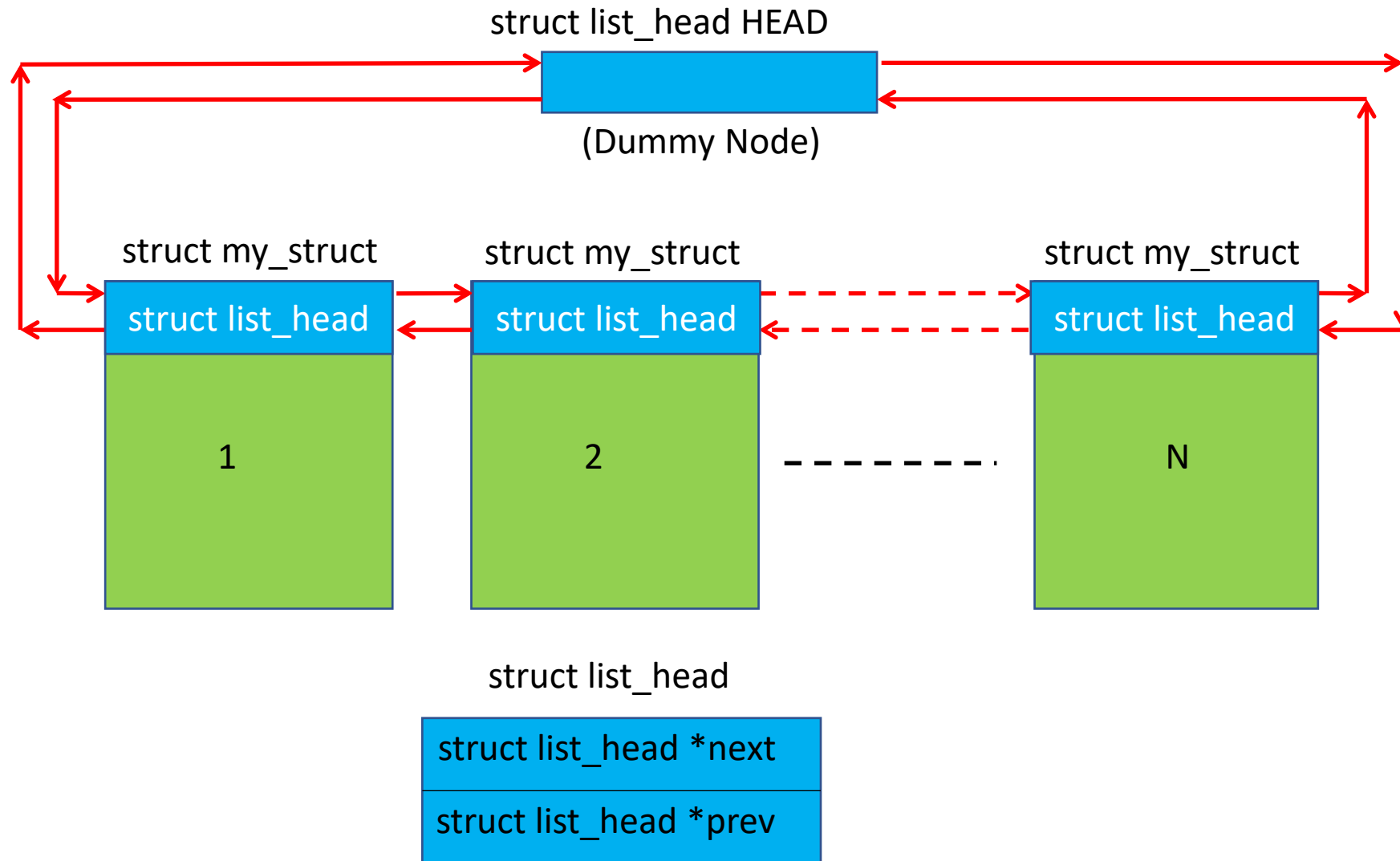
Traditional Circular Doubly-Linked List - 1



Traditional Circular Doubly-Linked List - 2



Circular Doubly-Linked List using struct list_head



struct list_head - Definition

Header File: types.h

```
struct list_head
{
    struct list_head *next, *prev;
};
```

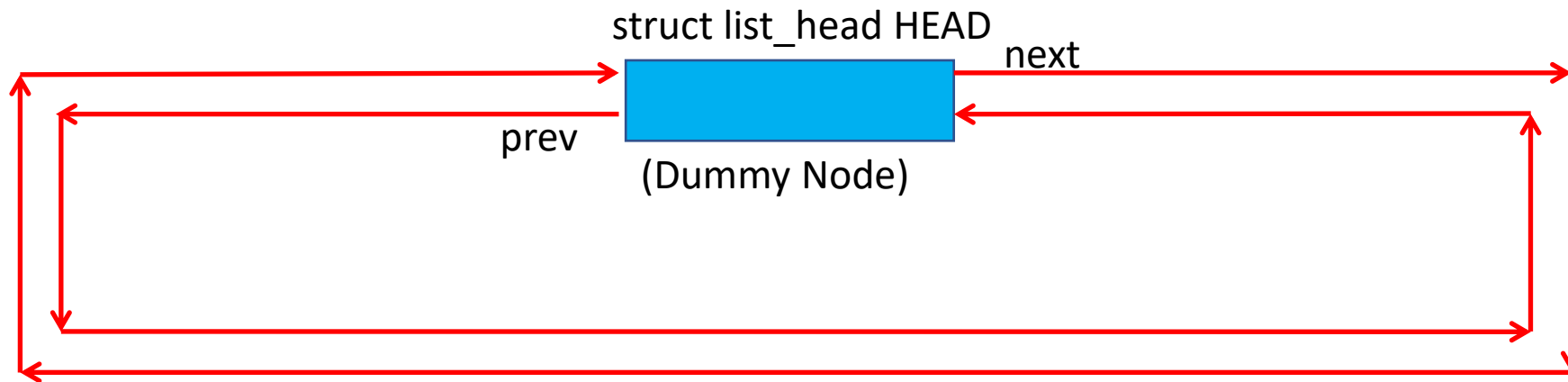
struct list_head - Initialization

Header File: list.h

```
void INIT_LIST_HEAD(struct list_head *list)
```

Example:

```
struct list_head HEAD;  
INIT_LIST_HEAD(&HEAD);
```



struct list_head – Add Operation-1

Header File: list.h

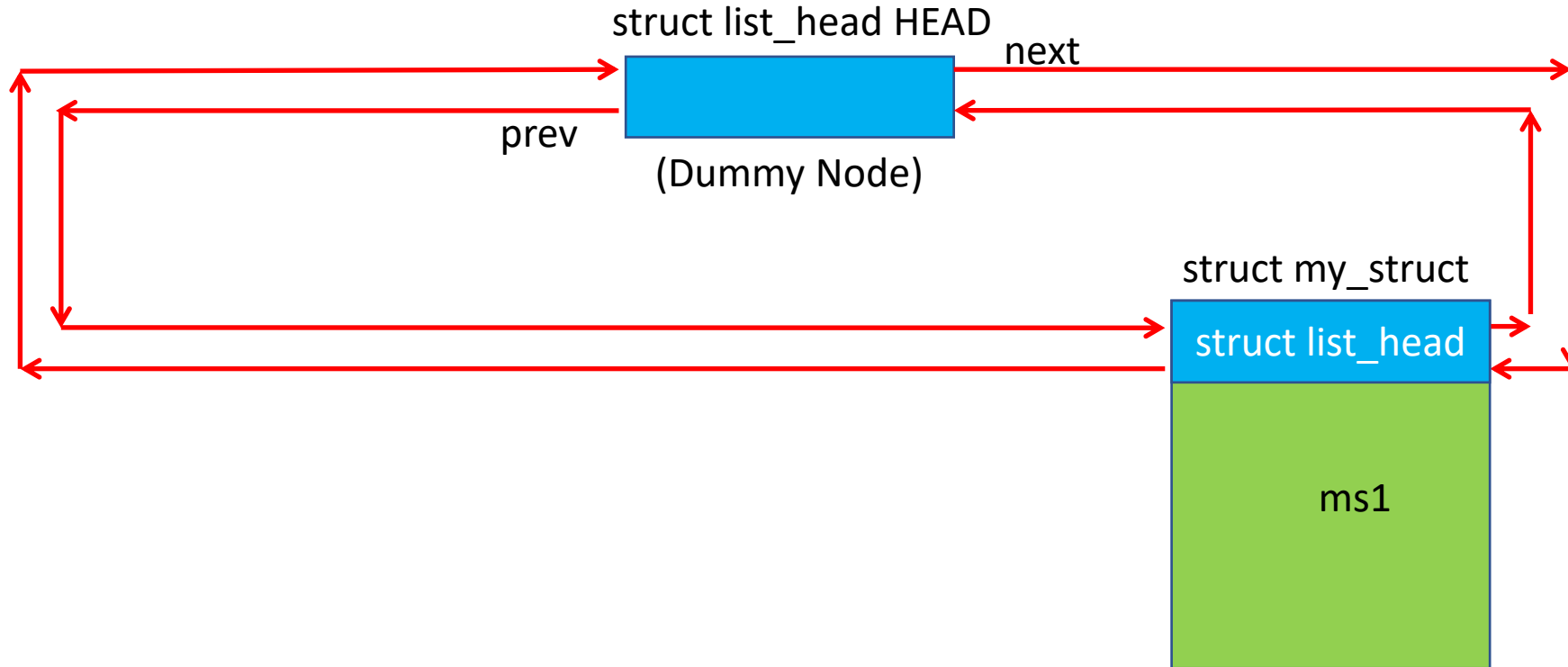
```
void list_add (struct list_head *new, struct list_head *head)
```

Example:

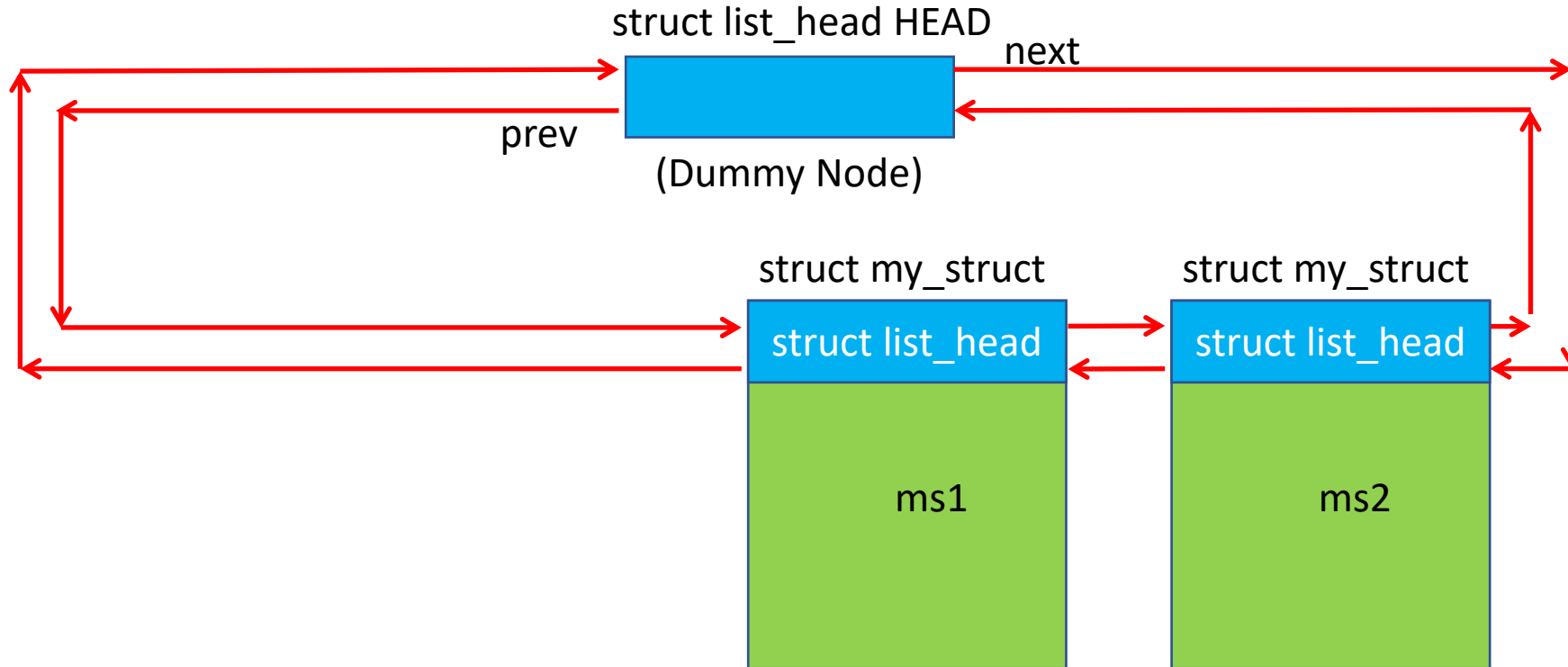
```
struct my_struct
{
    struct list_head link;
    -----
};

struct my_struct ms1, ms2;
list_add (&ms1.link, &HEAD);
list_add (&ms2.link, &HEAD);
```

struct list_head – Add Operation-2



struct list_head – Add Operation-3



struct list_head – Add at Tail - 1

Header File: list.h

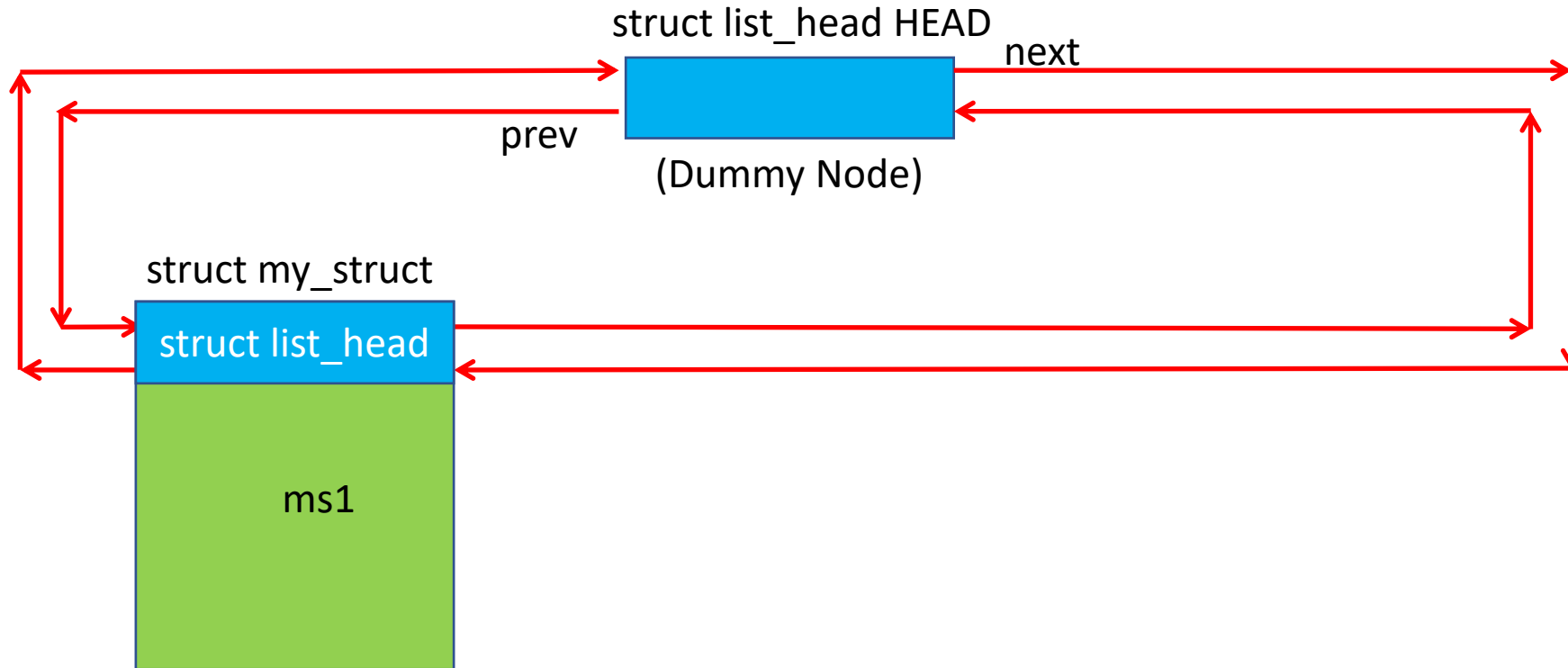
```
void list_add_tail (struct list_head *new, struct list_head *head)
```

Example:

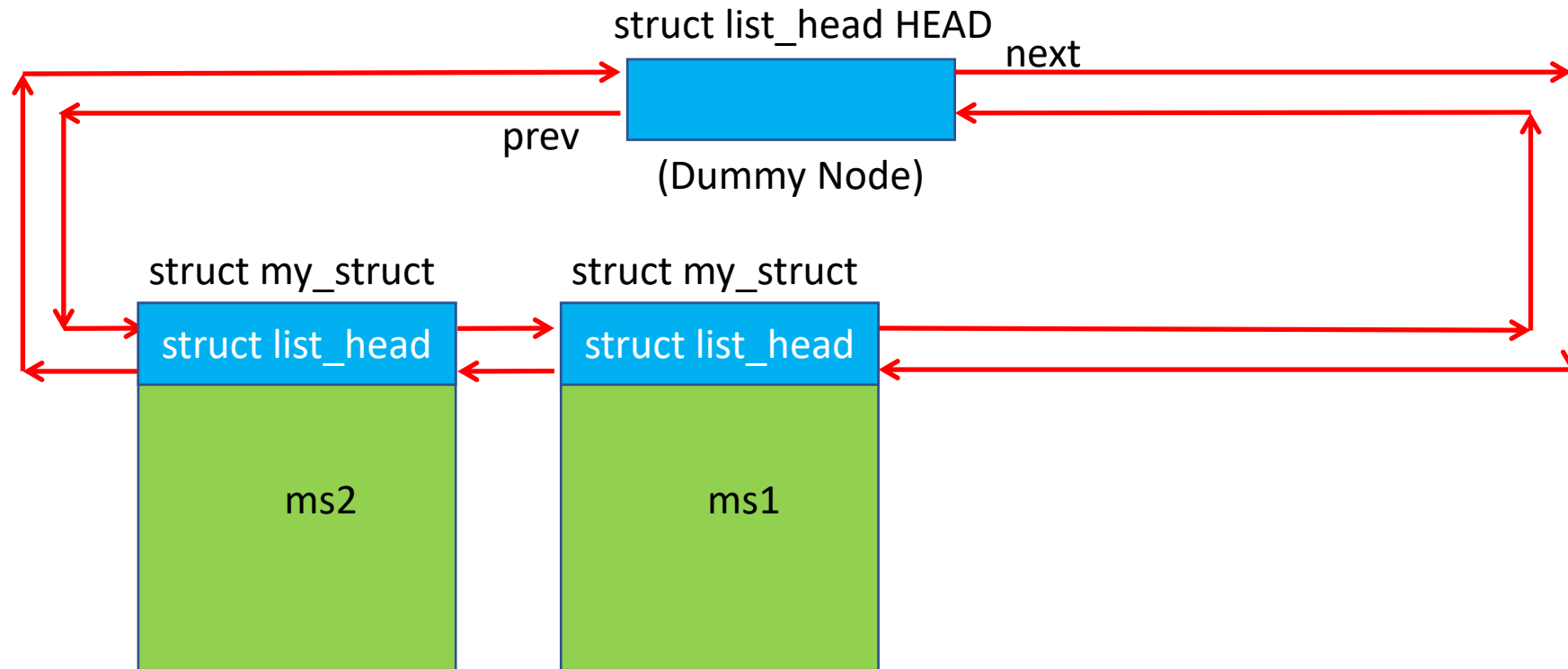
```
struct my_struct
{
    struct list_head link;
    -----
};

struct my_struct ms1, ms2;
list_add_tail (&ms1.link, &HEAD);
list_add_tail (&ms2.link, &HEAD);
```

struct list_head – Add at Tail - 2



struct list_head – Add at Tail - 3



struct list_head – Delete Operation-1

Header File: list.h

```
void list_del (struct list_head *link)
```

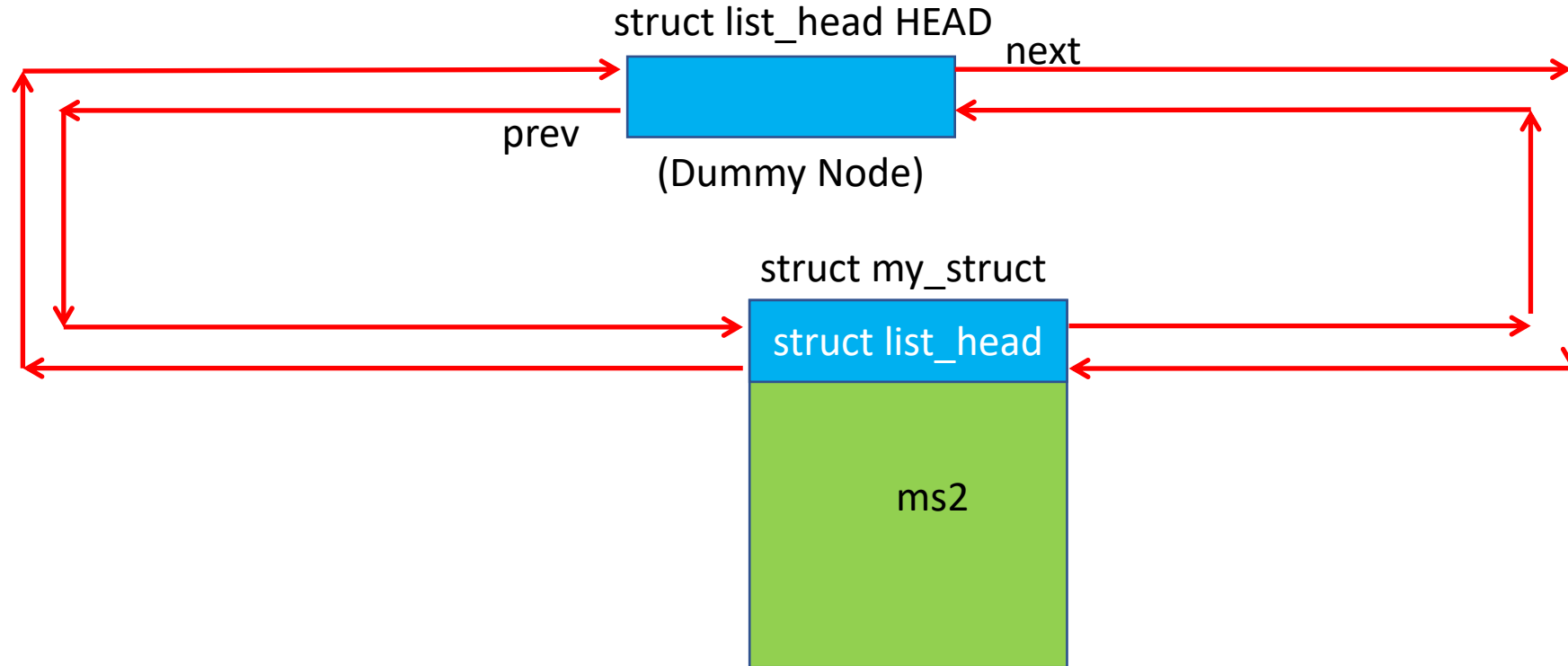
Example:

```
struct my_struct ms1, ms2;
```

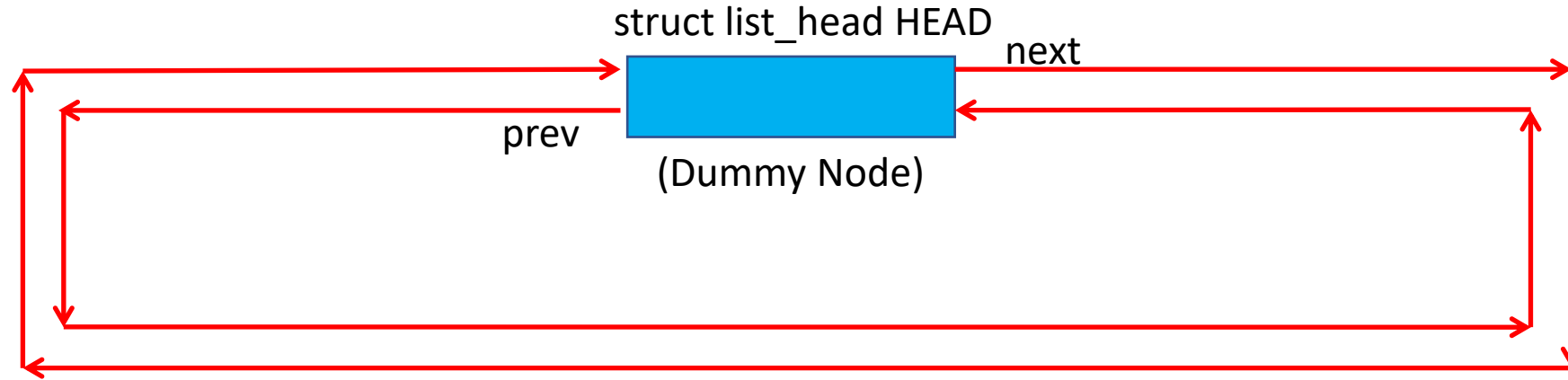
```
list_del (&ms1.link);
```

```
list_del (&ms2.link);
```

struct list_head – Delete Operation-2



struct list_head – Delete Operation-3



struct list_head – Traversal Operation

Header File: list.h

Macro:

`list_for_each_entry (pointer-to-my-struct, pointer-to-head, link_member_name)`

Example:

```
// This code deletes each entry while traversing through the list
struct my_struct
{
    struct list_head link;
    -----
};
struct my_struct *entry;
list_for_each_entry (entry, &HEAD, link)
{
    list_del(&entry->link);
}
```


struct list_head – Misc. Operation

Header File: list.h

```
int list_empty (struct list_head *head)
```

Example:

```
// Returns 1 (for empty list) or 0 (for non-empty list)
```

```
list_empty (&HEAD);
```