Nginx Queues

Table of contents :

Introduction :

The concept of Queues is implemented in nginx using these two files :

- ngx_queue.c -> Present in src/core/ directory
- ngx_queue.h -> Present in src/core/ directory

In nginx the queues are mainly used to bind or chain the buffers in a sequence so that we know the order in which data is received,processed and sent.The data can't be processed until it is received and it can't be send until it is processed.The order and sync of these processes is important.

Queue Structure :

Nginx implements a double ended queue.
The data structure for the queue is defined as follows :

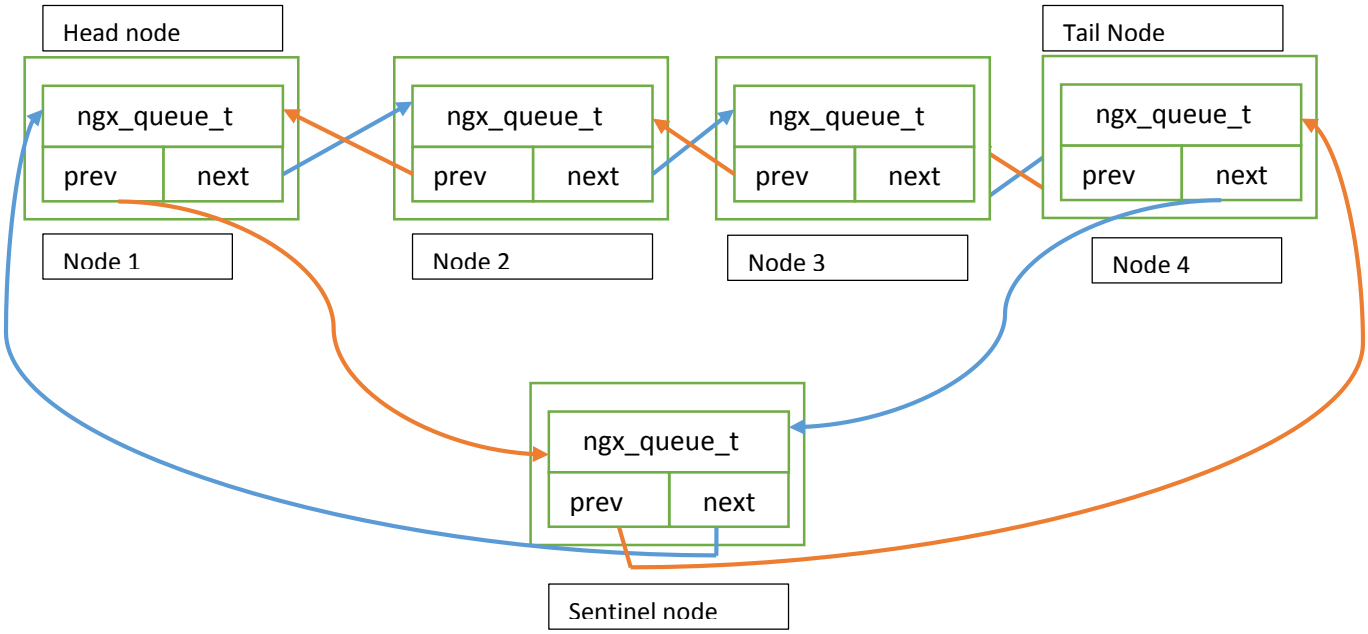
```
typedef struct ngx_queue_s ngx_queue_t;

struct ngx_queue_s {
    ngx_queue_t * prev;
    ngx_queue_t * next;
};
```

The queue structure provides two pointers prev and next.The prev pointer is a pointer to queue structure which points to the previous node in the list and similiarly next points to the next node in the list.From this structure we can observe that the queue structure provides no field for storing the data.If we need to store the data we need to define our own structure accordingly. Eg : For managing the student Info Like USN,Name,Sex,age we need to define the queue as :

```
typedef struct studentInfo{
        ngx_uint_t  USN;
        u_char      *Name;
        char        sex;
        ngx_uint_t  age;
        ngx_queue_t queue;
}student;
```

Queue Implementation:

The queue is Double ended.There is a head node and a tail node which represent the start and end of the queue.Apart from these two there is a sentinel node which points to both the head and tail.And both head and tail point to this sentinel node.
This sentinel node acts as a traversal path terminator. Sentinel node is an alternative to NULL pointer.The main advantage of using sentinel over NULL is efficiency and faster searching or traversing.This is because we need not to check for null value everytime to access the node.Skipping this check saves a lot of time.The sentinel node does not store or reference any data.


Head node
Tail Node
ngx_queue_t
prev
next
Node 1
Node 2
Node 3
Node 4
Sentinel node