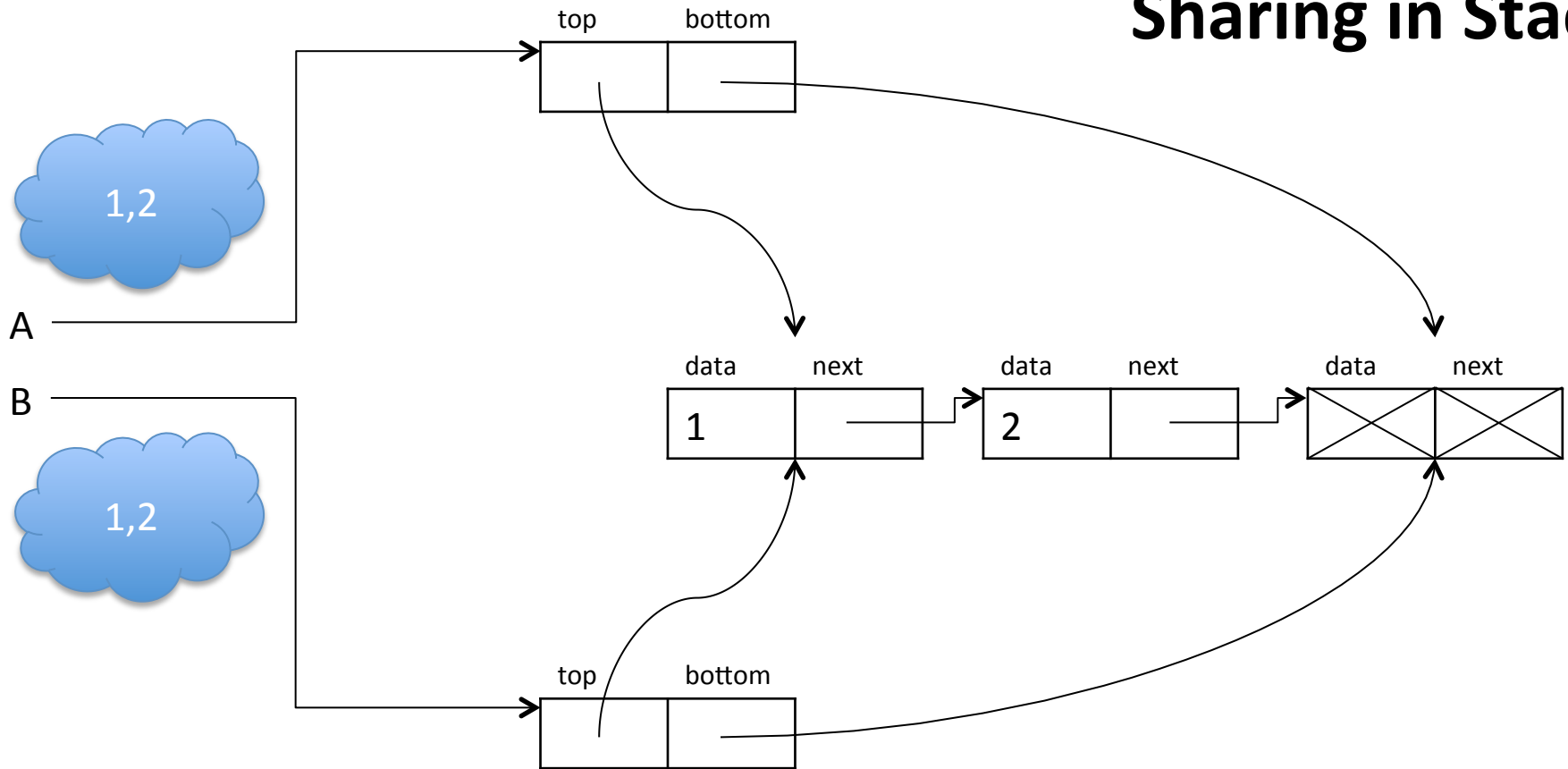
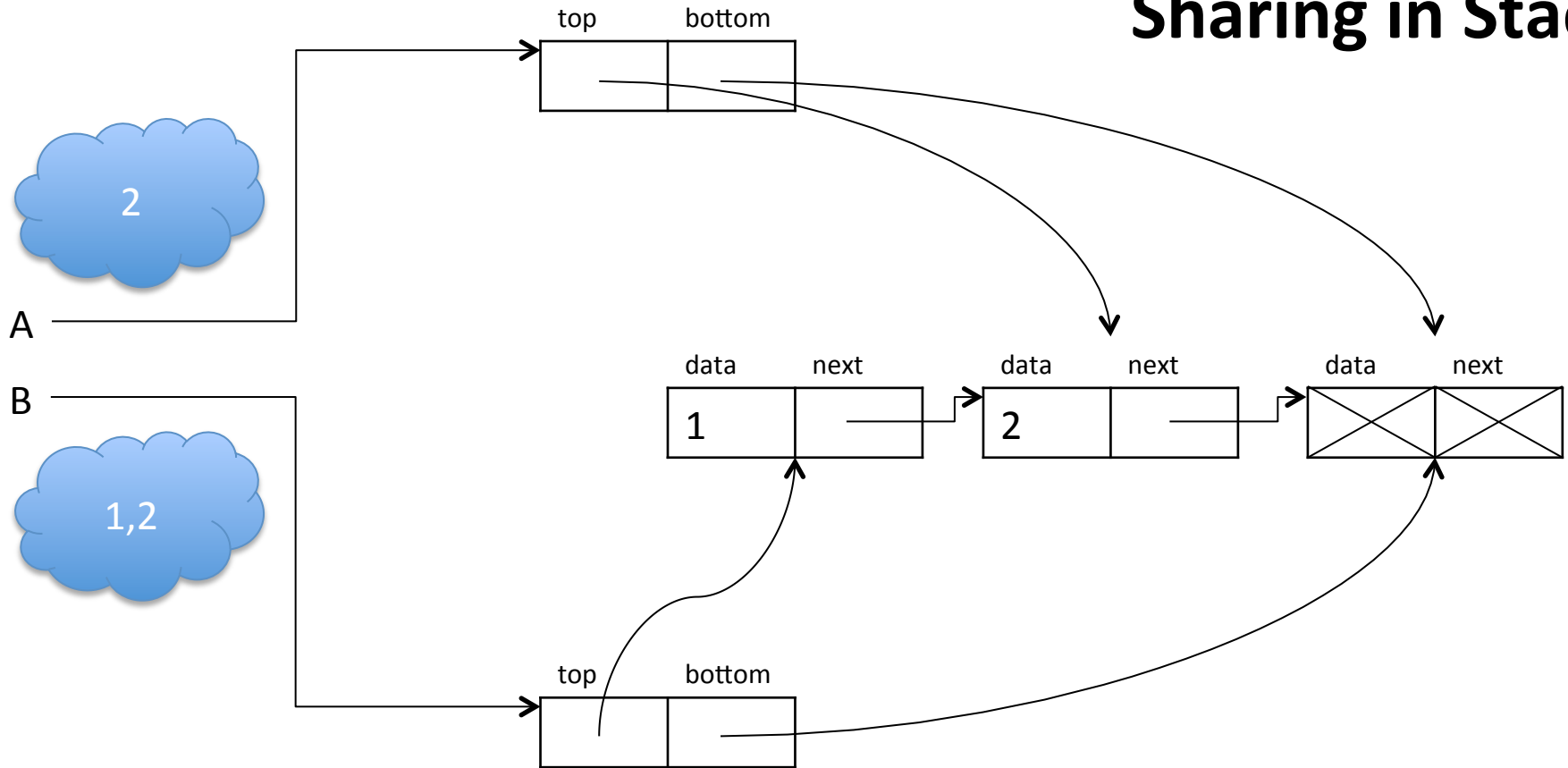


Sharing in Stacks



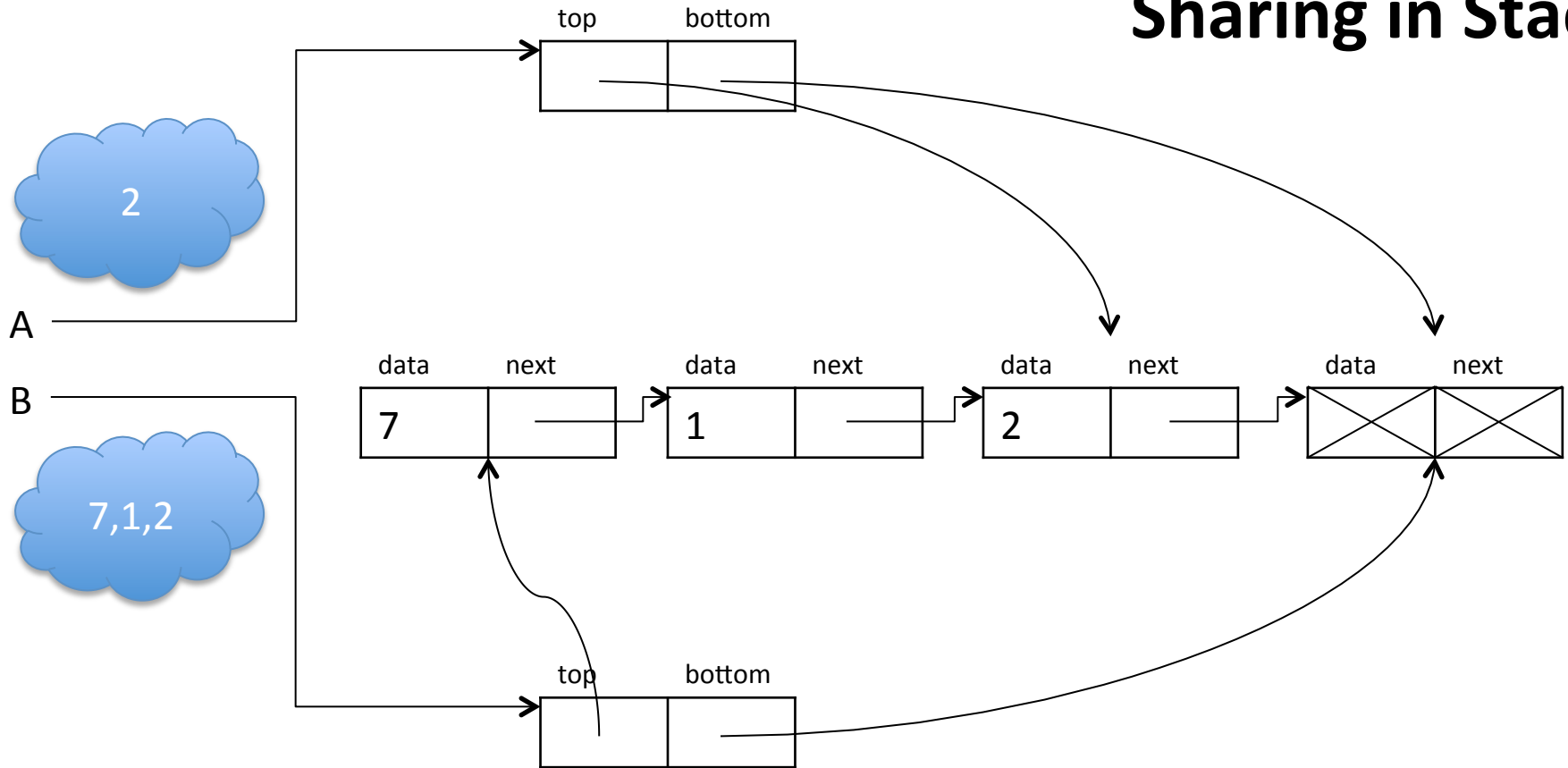
```
pop(A);  
push(B, 7);  
push(A, 3);  
pop(B);
```

Sharing in Stacks



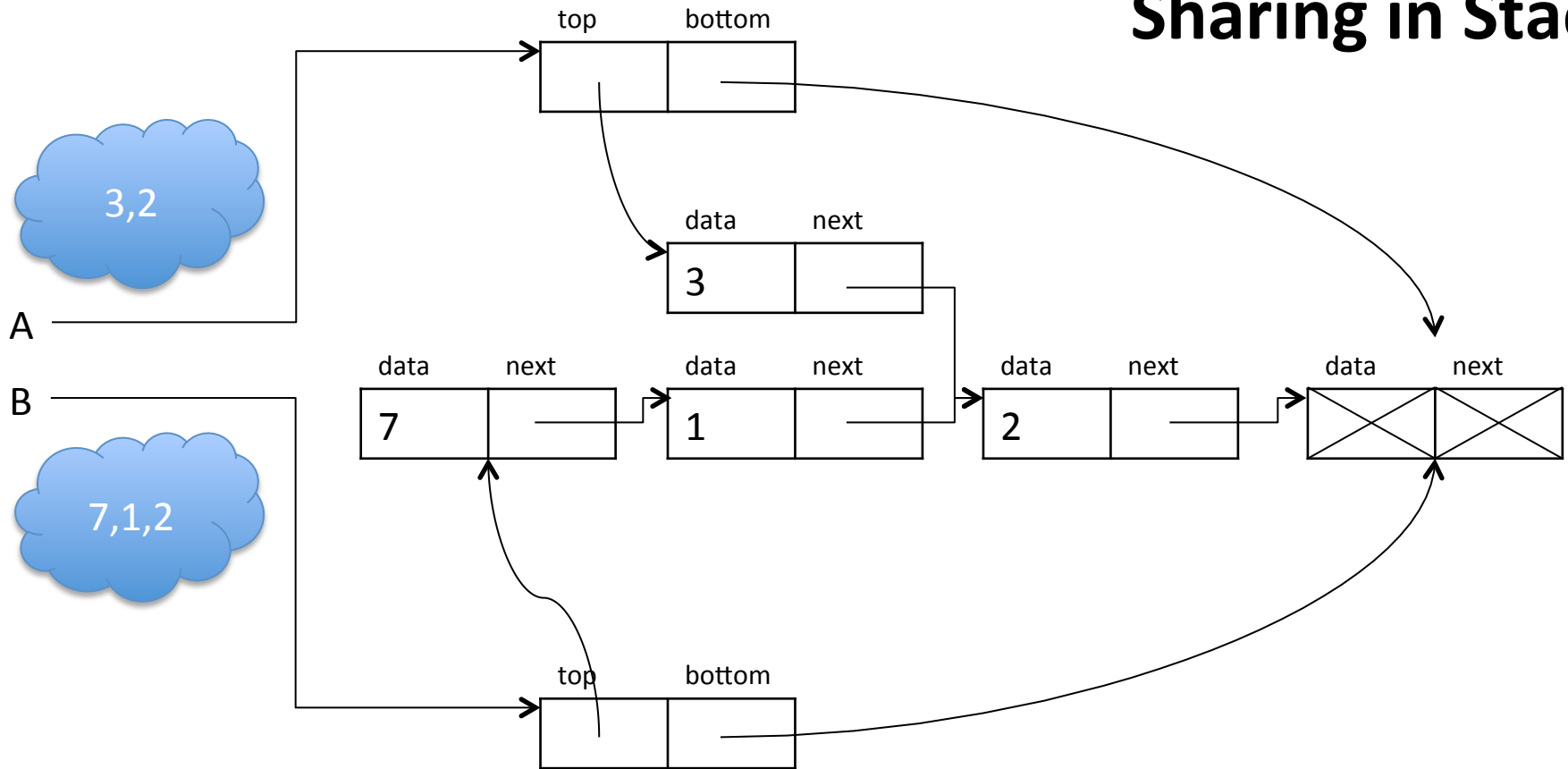
```
pop(A) ;  
push(B, 7) ;  
push(A, 3) ;  
pop(B) ;
```

Sharing in Stacks



 **pop (A) ;**
push (B, 7) ;
push (A, 3) ;
pop (B) ;

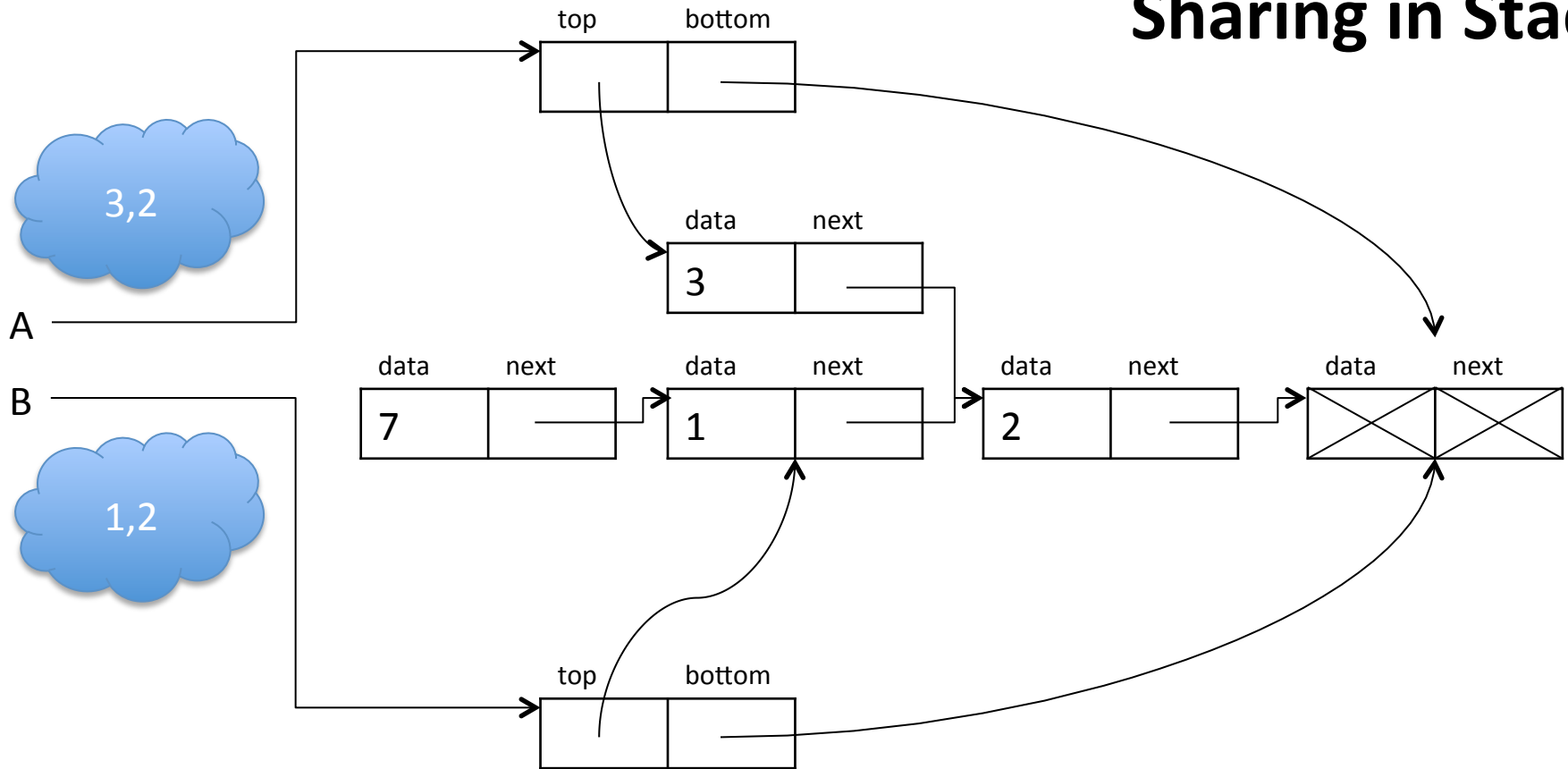
Sharing in Stacks



pop (A) ;
push (B, 7) ;
push (A, 3) ;
pop (B) ;



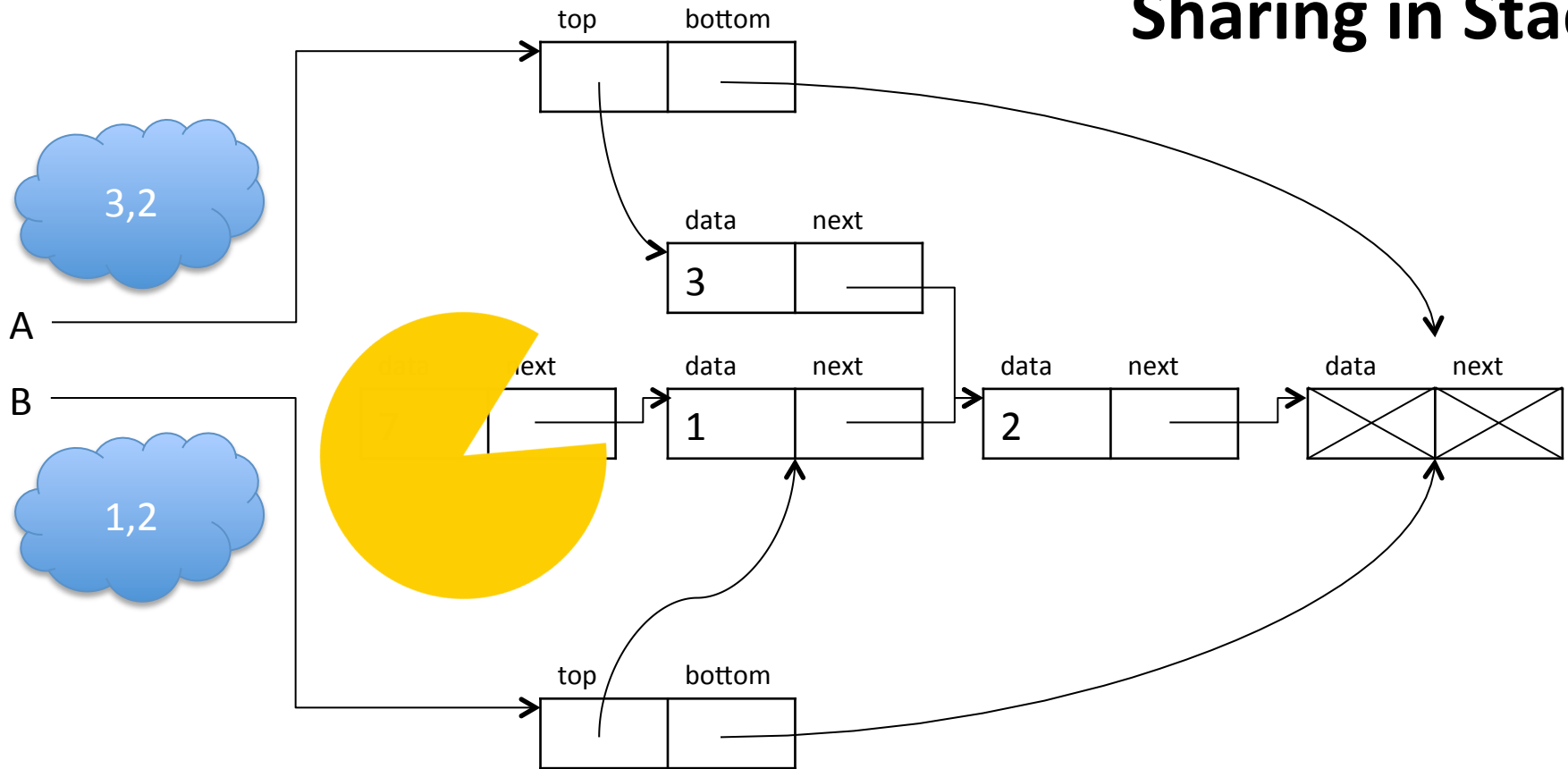
Sharing in Stacks



```
pop (A) ;  
push (B, 7) ;  
push (A, 3) ;  
pop (B) ;
```



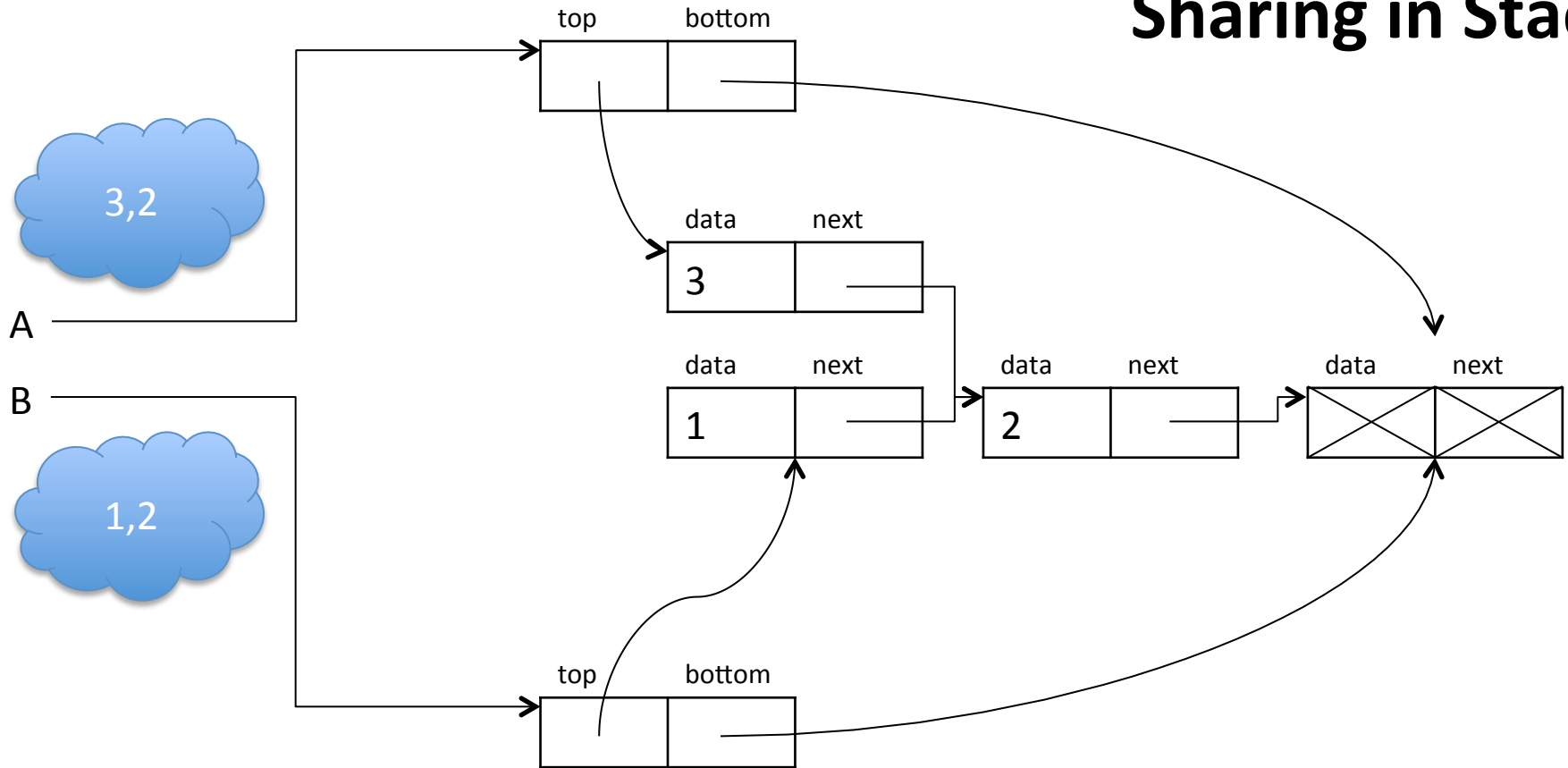
Sharing in Stacks



```
pop (A) ;  
push (B, 7) ;  
push (A, 3) ;  
pop (B) ;
```



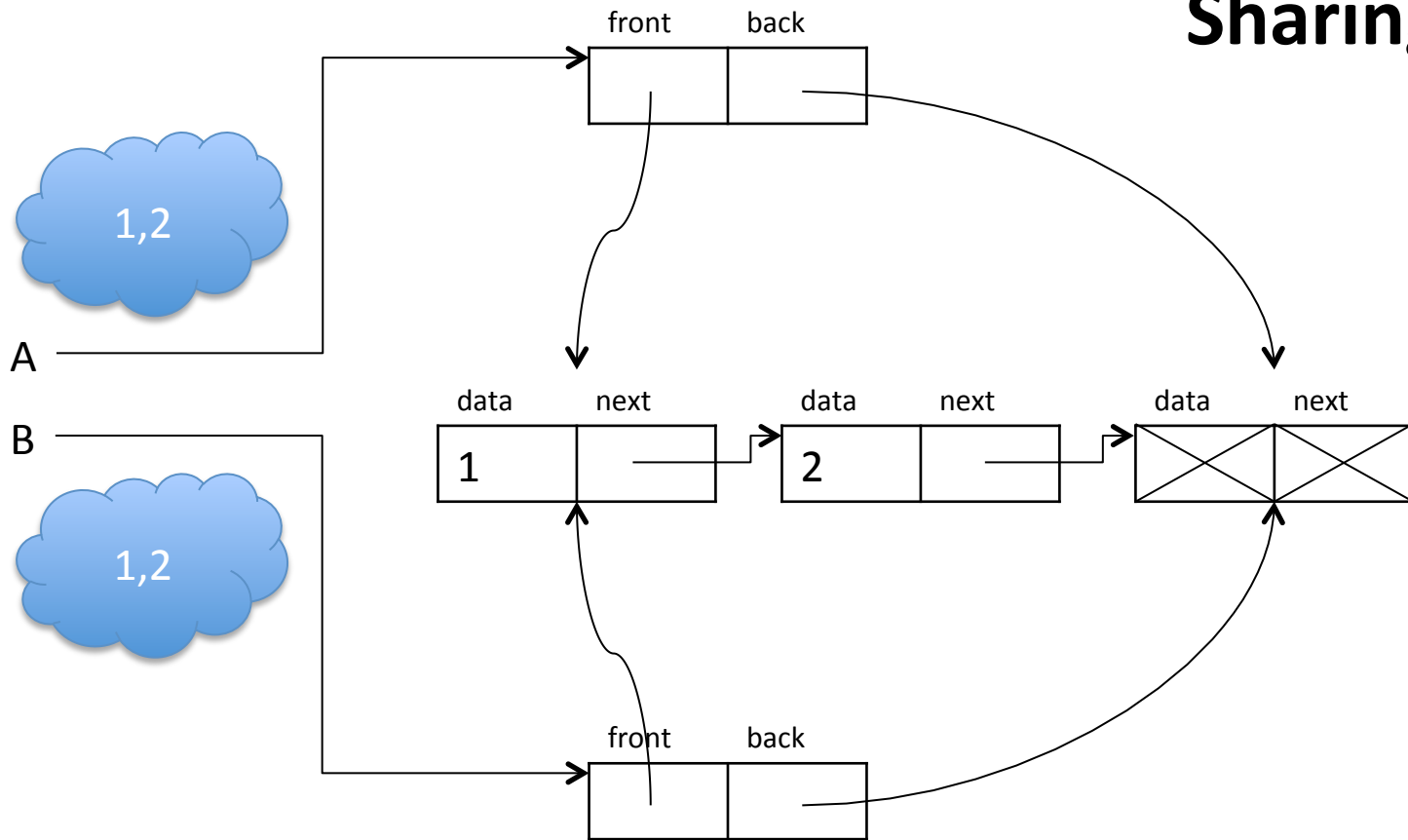
Sharing in Stacks



```
pop (A) ;  
push (B, 7) ;  
push (A, 3) ;  
pop (B) ;
```

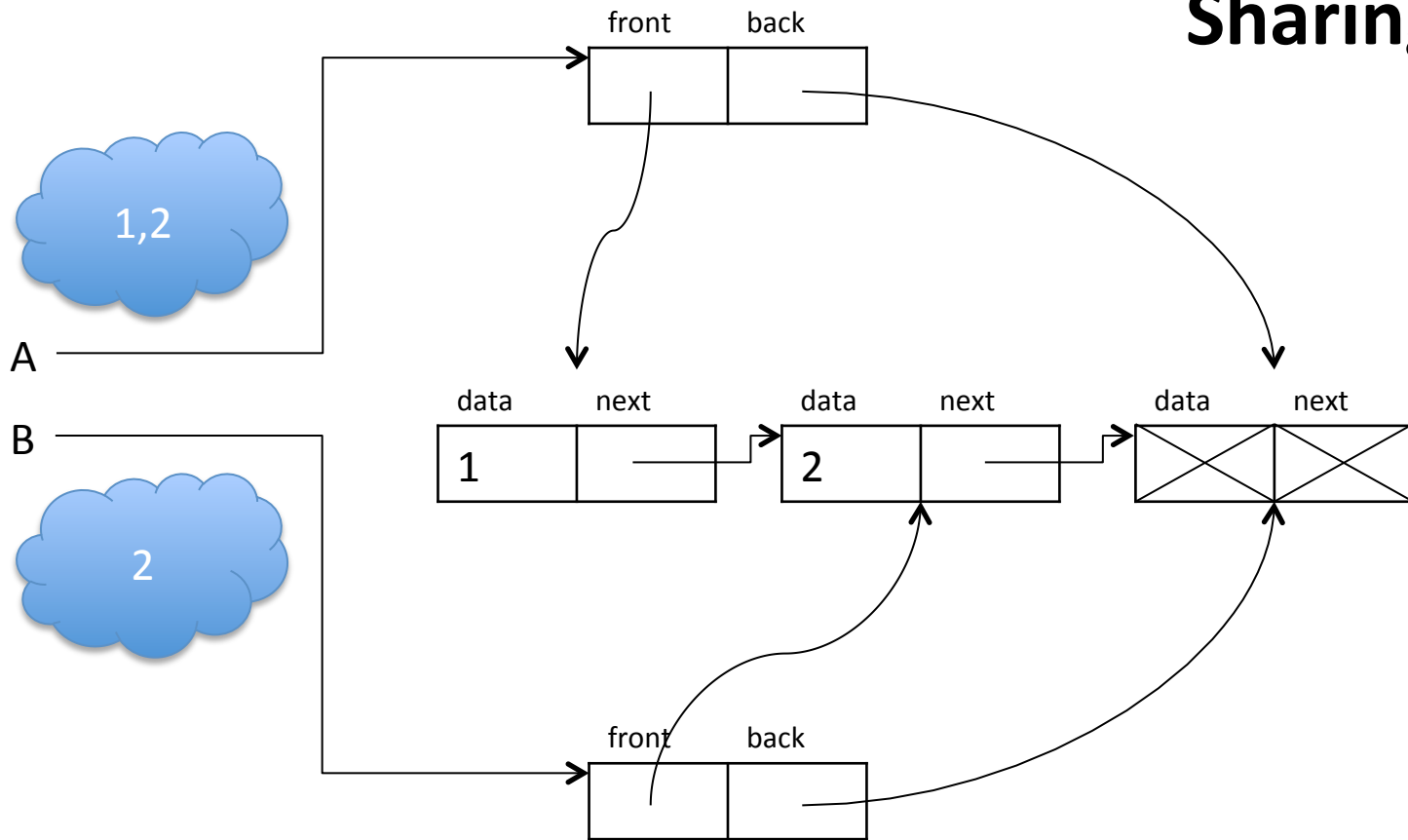


Sharing in Queues



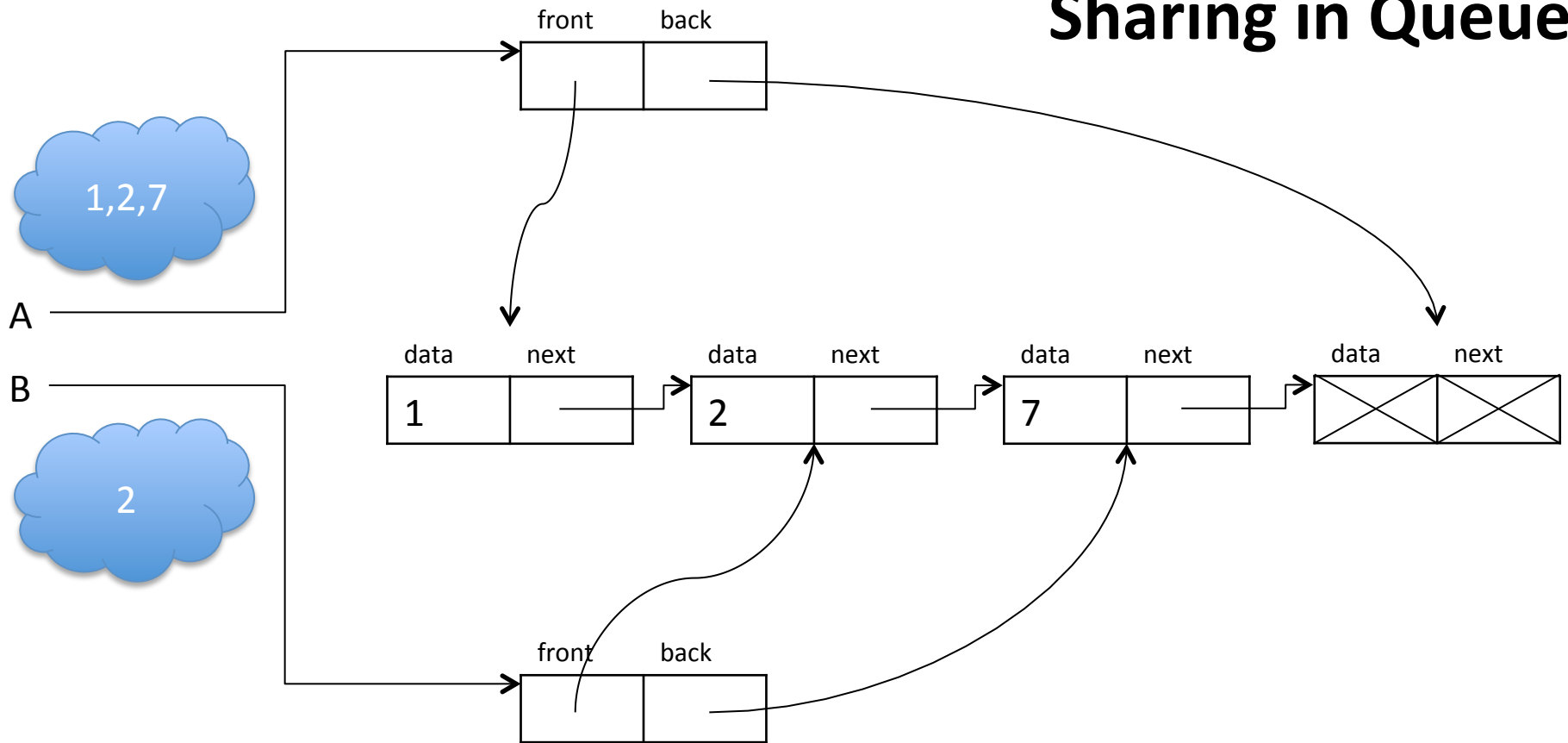
```
deq(B) ;  
enq(A, 7) ;  
deq(A) ;  
enq(B, 3) ;
```


Sharing in Queues



```
deq(B) ;  
enq(A, 7) ;  
deq(A) ;  
enq(B, 3) ;
```

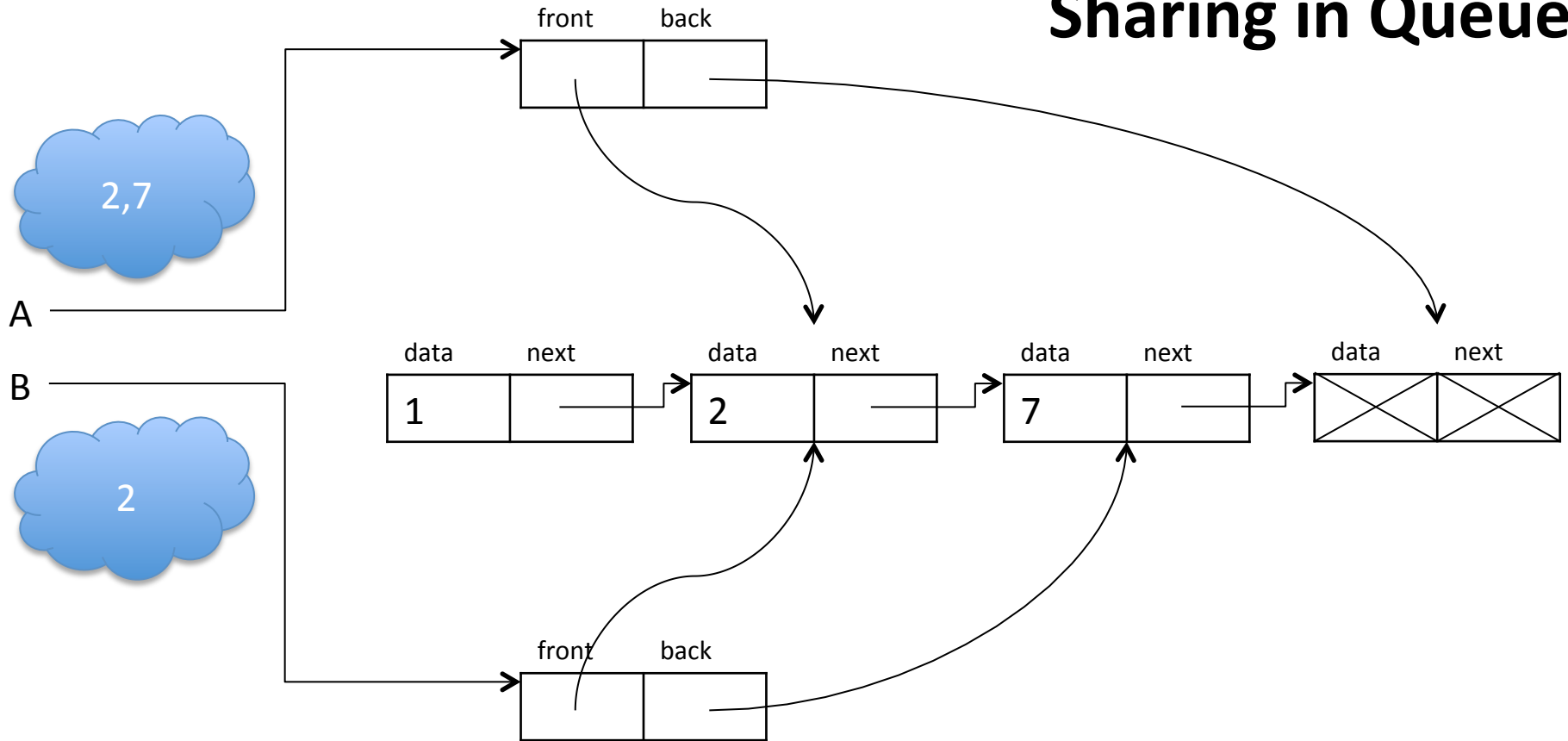
Sharing in Queues



deq (B) ;
enq (A, 7) ;
deq (A) ;
enq (B, 3) ;



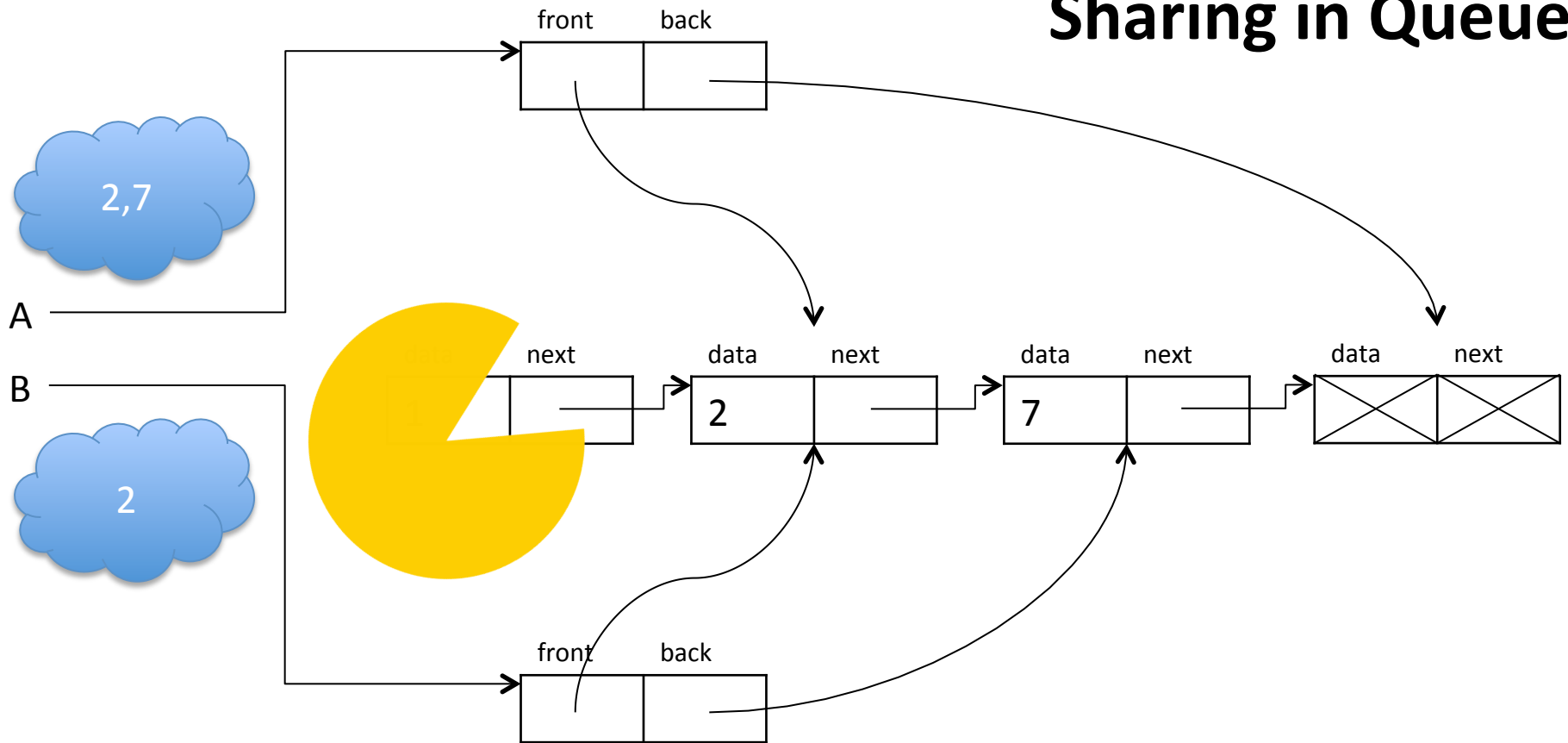
Sharing in Queues



deq (B) ;
enq (A, 7) ;
deq (A) ;
enq (B, 3) ;



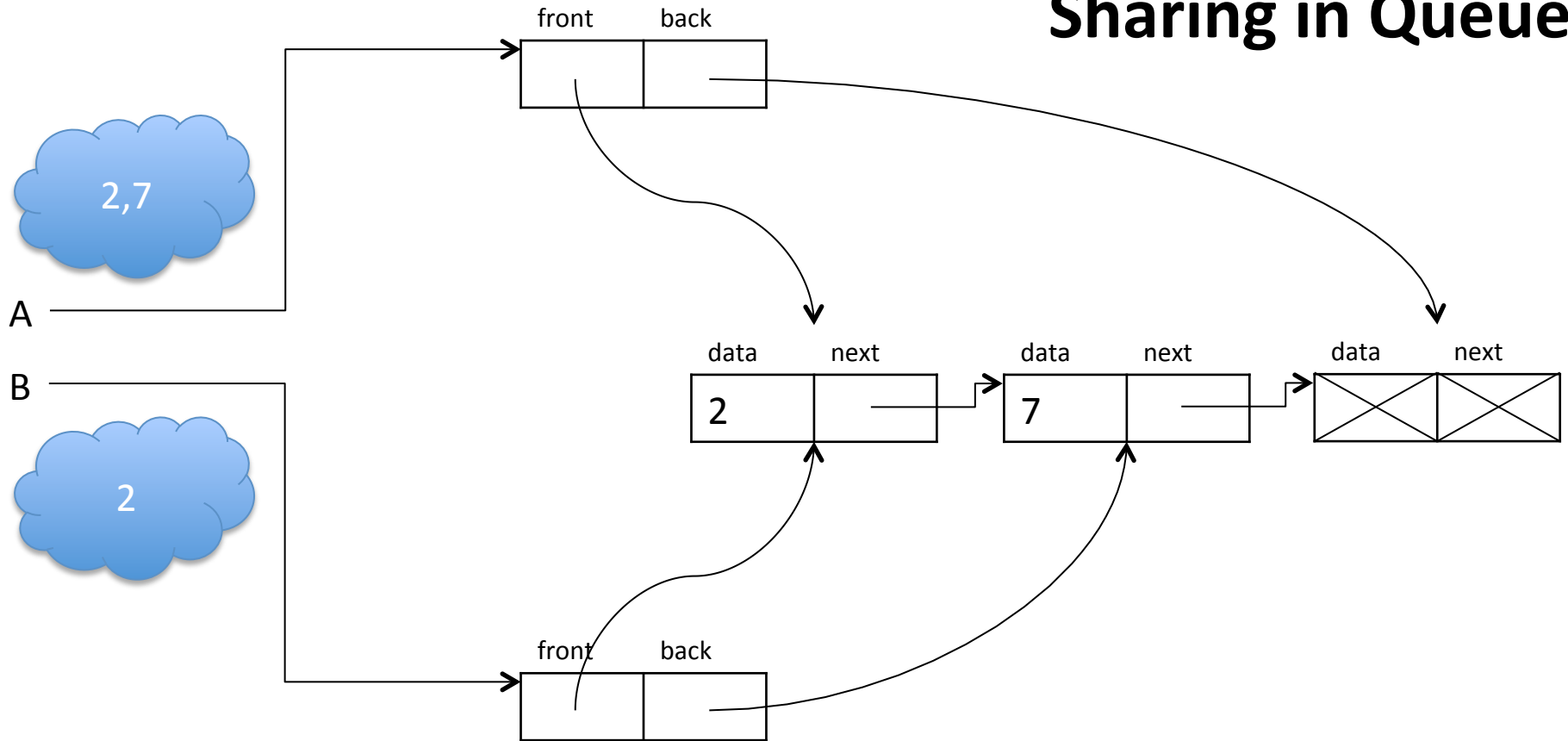
Sharing in Queues



```
deq(B) ;  
enq(A, 7) ;  
deq(A) ;  
enq(B, 3) ;
```



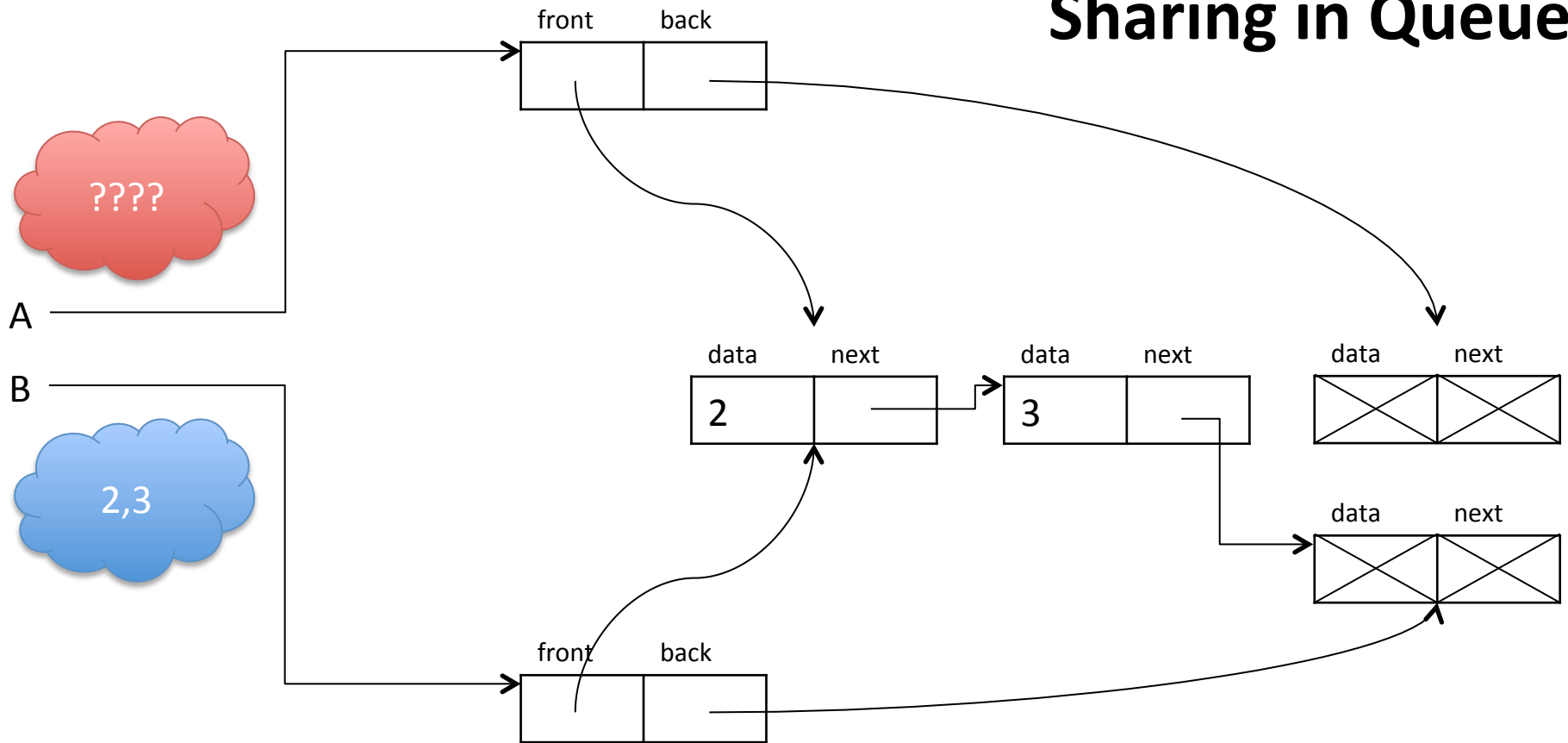
Sharing in Queues



deq (B) ;
enq (A, 7) ;
deq (A) ;
enq (B, 3) ;



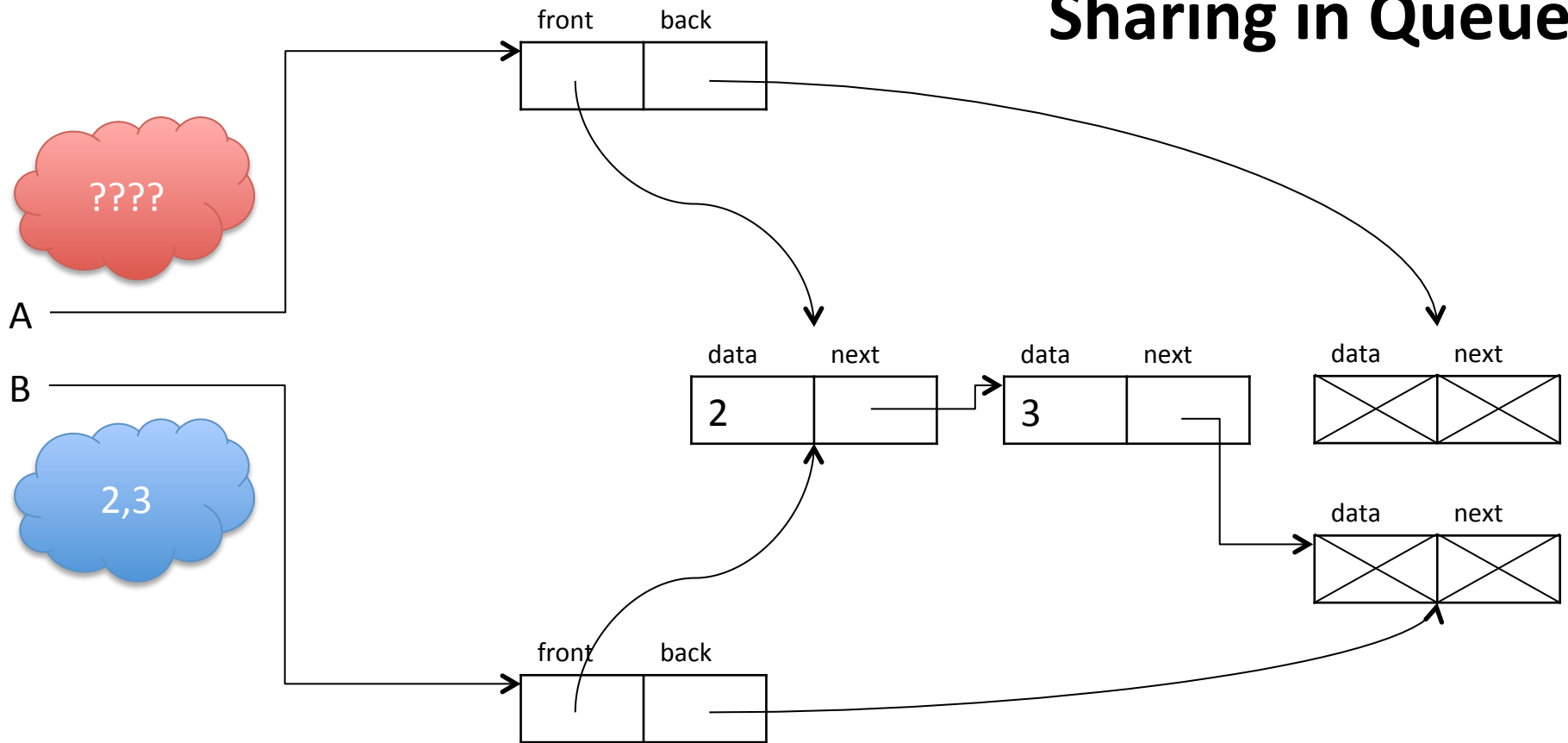
Sharing in Queues



```
deq (B) ;  
enq (A, 7) ;  
deq (A) ;  
enq (B, 3) ;
```



Sharing in Queues



```
deq(B) ;  
enq(A, 7) ;  
deq(A) ;
```

```
// enq(B, 3) ;
```



```
// We can enqueue/dequeue from A, but B must be 'READ ONLY'
```