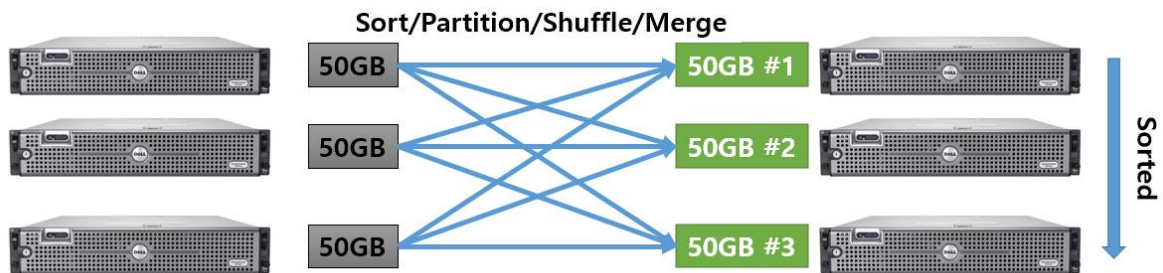


Algorithm for distributed/parallel sorting

project에서 input data의 크기는 memory, disk의 크기보다 훨씬 크다. Multiple machine에 들어온 input을 disk단위로 정렬하여 multiple machine에 저장해야 한다.

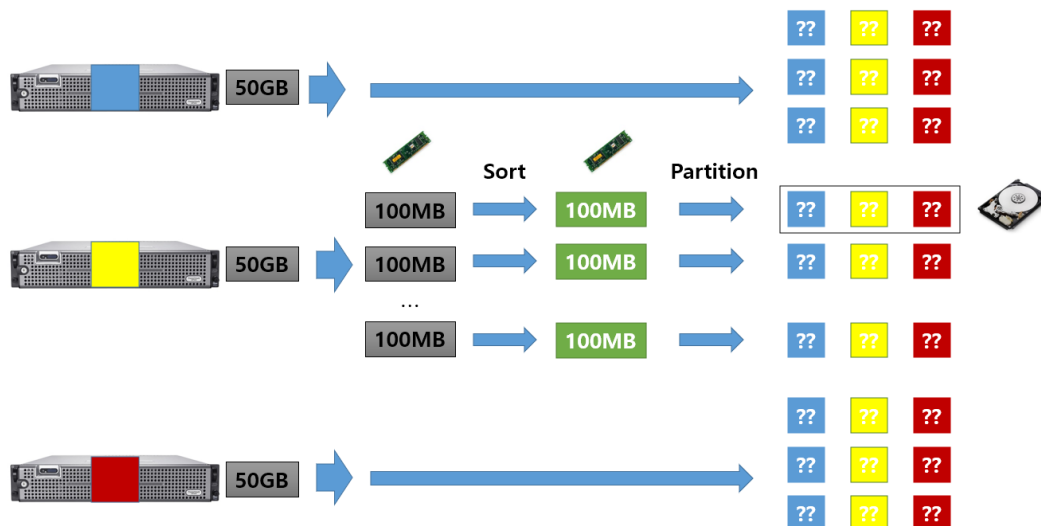
disk단위로 distributed/parallel sorting algorithm에 대해 알아보았다.

설명하기 쉽게 강의 ppt에 나와있는 사진들을 사용하였다.



위 그림에서는 총 용량이 50GB인 3개의 worker machine에 input이 들어와 정렬한 후 차례대로 worker machine에 저장하는 형태인 것을 볼 수 있다.

Parallel sorting을 위해서 sort->sampling->partition->shuffle->merge 순서를 따라서 진행하면 될 것 같다.

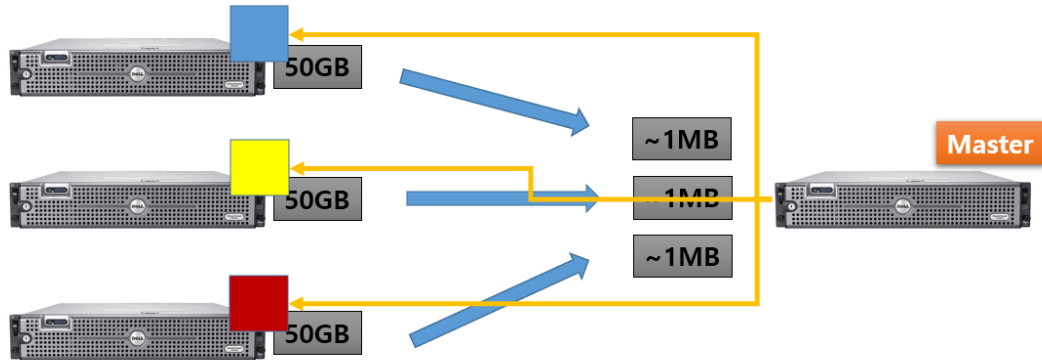


위 그림에서는 worker machine의 disk 크기가 각 100MB라고 하였다.

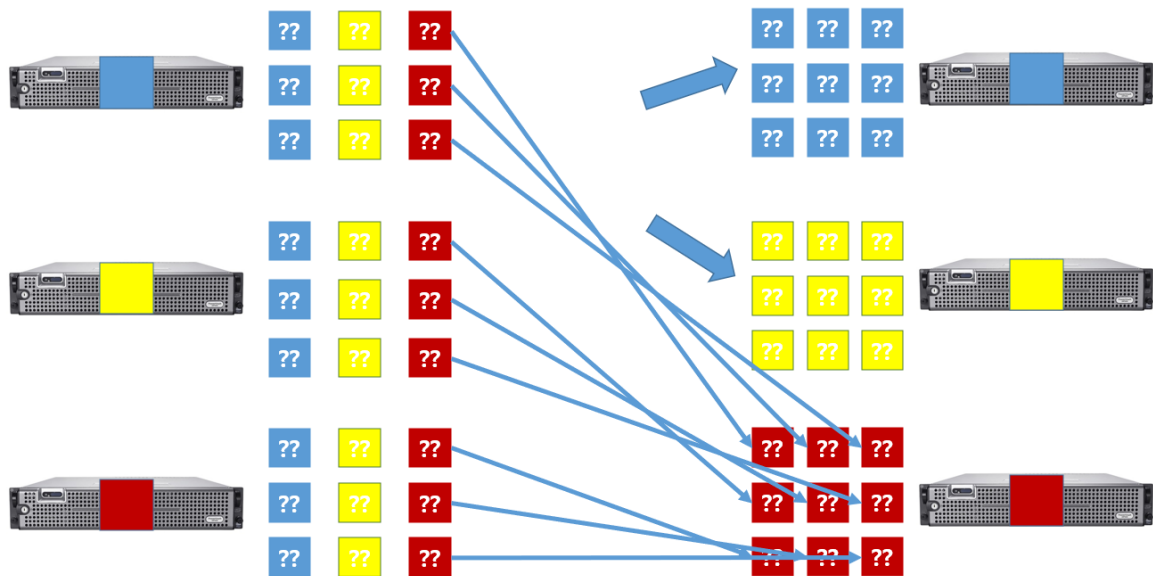
첫번째로 disk에 있는 input 데이터를 정렬한다. 이때 quick sort를 이용해 진행한다.

이후 어떤 값을 기준으로 sector를 나눈다.

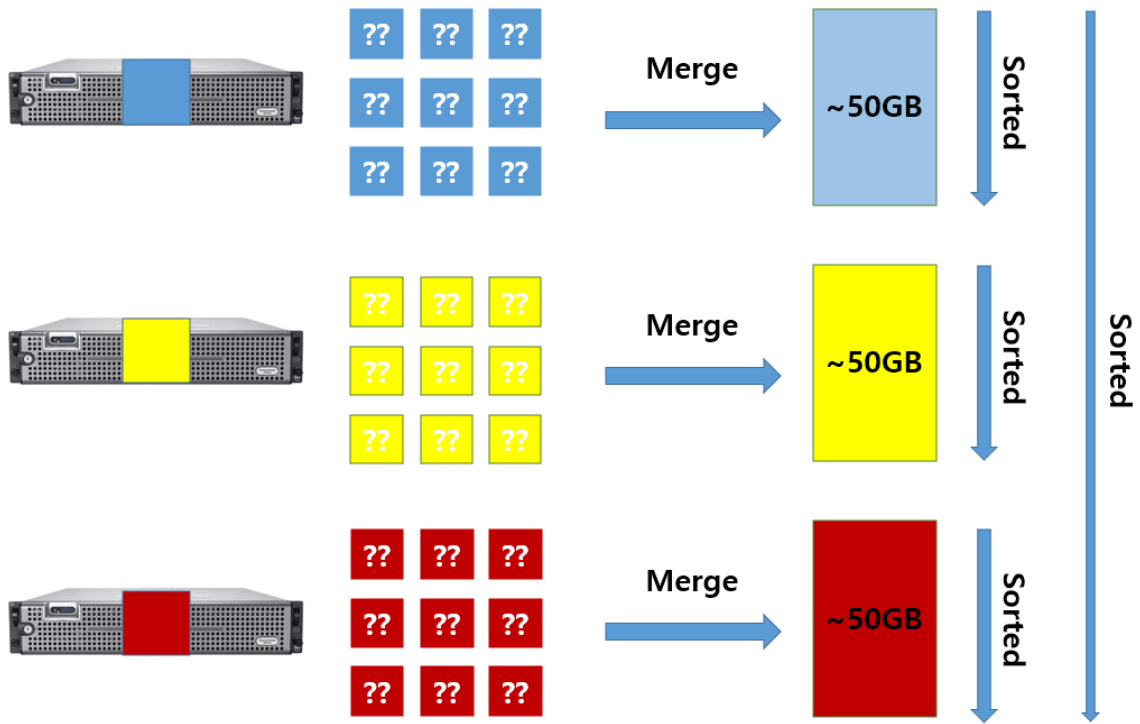
n개의 data를 p개의 worker machine으로 나눈다고 하면 p개의 partition으로 나눈다. 이때 값을 나누는 값은 master machine을 이용한다.



각 disk에 존재하는 데이터의 수가 N개라고 하면 $0, \frac{N}{p}, \frac{2N}{p}, \dots, \frac{(p-1)N}{p}$ 번째 data들을 모아서 Master machine에 전달한다. 그러면 master machine에서는 data들을 정렬하여 (이때도 quick sort를 쓰면 좋을 것 같다) p-1개의 pivot data를 각 worker machine에 전달한다. p-1개의 pivot data를 전달받은 worker machine의 각 disk에서는 p개의 sector로 구분한다.



p개의 sector로 나누면 각 sector가 동일한 machine에 저장될 수 있도록 shuffle한다.



이후 각 worker machine에는 p 개의 정렬되어 있는 sector로 이루어져 있다. 각 Worker machine에서 p 개의 정렬되어 있는 목록을 하나로 병합한다.

최종적으로 n 개의 data가 p 개의 processor에 정렬되어 있는 상태로 저장된다.