DISTRIBUTED INDEXING

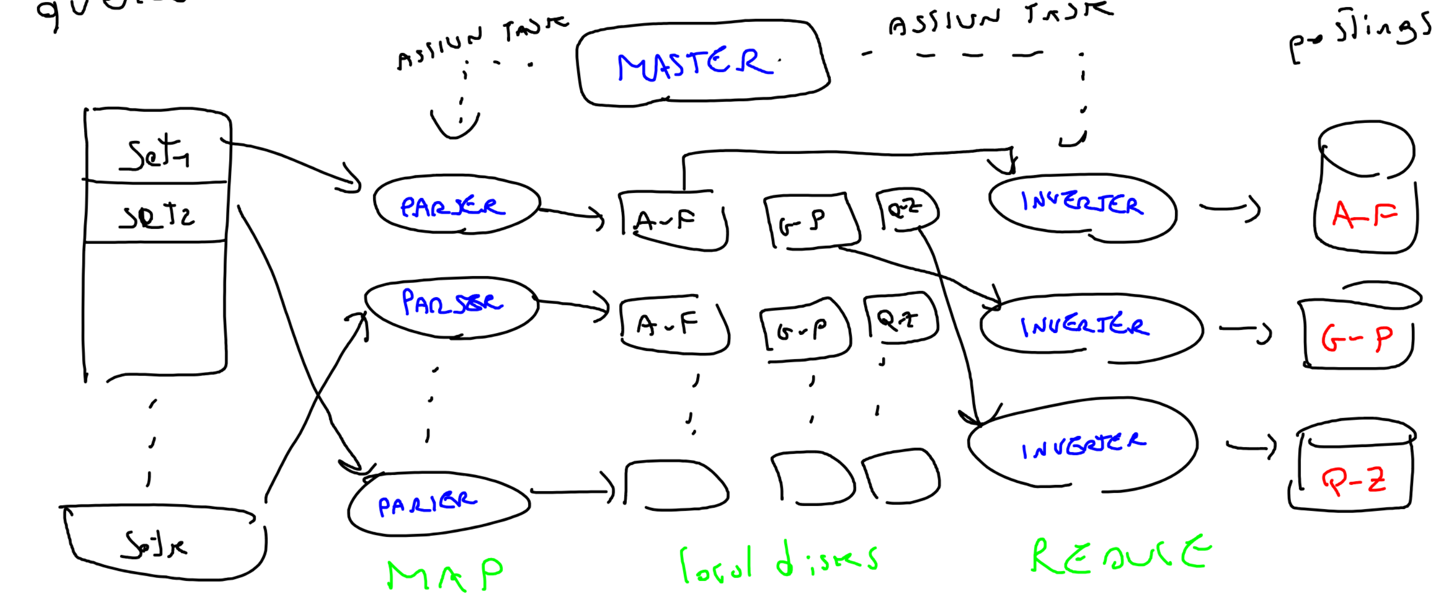
For a web-scale indexing, it is important to balance the workload between different machines, due to the massive size and the fault-prone nature of a single machine:

1) Divide the indexing into a set of parallel tasks: the parsing task and the inverting task
2) A master machine assign each task to a pool of slave machines

There are two types of partitioning, executed by the parsing machines:

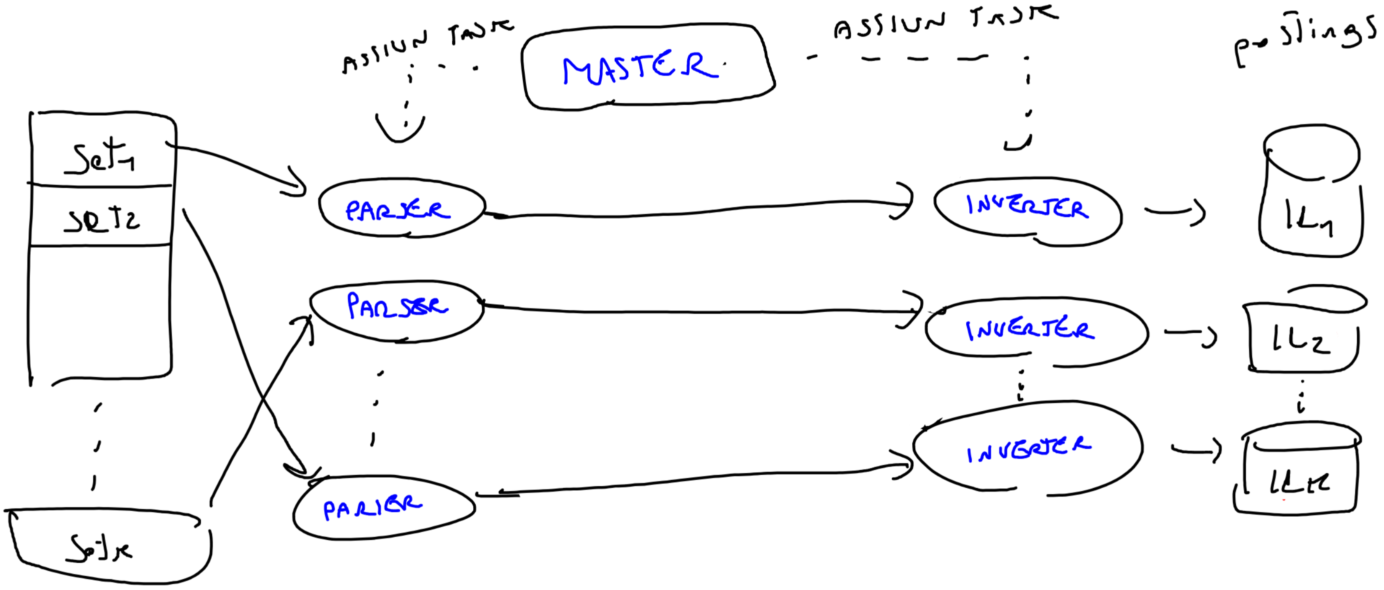
Term-based partitioning

Each slave machine contains a partitions of terms and their corresponding posting lists. Therefore, the queries are routed to the correct inverter machine.



DOC-BASED PARTITIONING

Each slave machine contains the inverted index of a subset of documents. Therefore, the queries go to all inverted machines, and the result presented to the user may be the join of results coming from multiple machines.



DOC-BASED VS TERM-BASED

Term-based allows for greater in concurrency, but in practice it's difficult to ensure good load balancing and to allow for multi word queries.

Doc-based makes the latter problem less difficult, but requires a global operation at each query