To facilitate big data processing, many distributed analytic frameworks and storage systems such as Apache Hadoop, Apache Hama, Apache Spark and Hadoop Distributed File System (HDFS) have been developed. Currently, many researchers are conducting research to either make them more scalable or enable them to support more analysis applications. In my PhD study, I conducted three main works in this topic, which are minimizing the communication delay in Apache Hama, minimizing the memory space and computational overhead in HDFS and minimizing the disk I/O overhead for approximation applications in Hadoop ecosystem. Specifically, In Apache Hama, communication delay makes up a large percentage of the overall graph processing time. While most recent research has focused on reducing the number of network messages, we add a runtime communication and computation scheduler to overlap them as much as possible. As a result, communication delay can be mitigated. In HDFS, the block location table and its corresponding maintenance could occupy more than half of the memory space and 30% of processing capacity in master node, which severely limits the scalability and performance of master node. We propose Deister that uses deterministic mathematical calculations to eliminate the huge block location table and its corresponding maintenance. My third work proposes to enable both efficient and accurate approximations on arbitrary sub-datasets of a large dataset. Existing offline sampling based approximation systems are not adaptive to dynamic query workloads and online sampling based approximation systems suffer from low I/O efficiency and poor estimation accuracy. Therefore, we develop a distribution aware method called Sapprox. Our idea is to collect the occurrences of a sub-dataset at each logical partition of a dataset (storage distribution) in the distributed system at a very small cost, and make good use of such information to facilitate online sampling.