Abstract: In a distributed system that consists of independent servers organized in a cluster, consensus denotes the agreement of all servers on one shared state. Consensus algorithms can be applied to replicate state machines on several servers while guaranteeing resiliency to failures such as packet loss, network disconnects, or server failure. The consensus algorithm ensures that state machine changes are applied on all servers in the cluster that hold a copy of the state machine, or that they are rejected on all servers, that the state machines on all servers equal after the same changes have been applied and that the systems stays operational in certain failure scenarios.

In this talk I will give an introduction to the Raft algorithm for consensus in distributed systems which has gained a significant amount of attention despite being relatively new (published 2014). I will first give an overview of the problems that arise with state machine replication in distributed systems. After a short discussion of major practice-oriented consistency algorithms I will introduce basic concepts of the Raft algorithm, describe on how it achieves consensus in typical fault scenarios and briefly discuss the provability of the correctness of the algorithm. I will conclude this talk with a discussion on why RAFT has gained popularity in such short time.

Reference: