Summary
This paper puts forth a new algorithm for the distributed consensus problem. Distributed consensus usually becomes necessary when multiple machines maintain the same state and need to execute operations such that the overall system state remains consistent even if some of the servers are down. This requires a decision module which determines the order of operations to execute. The algorithm that decides the ordering of operations is called as the consensus algorithm.

Strong points and main contributions
Distributed consensus is solved by the PAXOS algorithm but is difficult to understand and reason about. The RAFT algorithm is designed with understandability as one of the primary design considerations together with correctness and safety. This is achieved by following two edicts:-

1. De-composing the problem and reasoning/solving for each problem separately.
2. State space reduction

Weak points
Since the approach is fairly new, several questions arise that have not been adequately answered in the paper. Some of these include:-

1. How does it behave for larger clusters? What are the leader election times in such cases? These details are not discussed in the evaluation section.
2. Strong leadership may be the cause of performance bottlenecks. How does it compare to PAXOS for large systems?
3. How are the leader terms decided - trade-off between time it takes for new leader to overwrite/replicate its log in a majority of servers vs. making steady progress.
4. Applicable for systems where broadcast mechanism is feasible as a way to communicate amongst multiple systems. Also, security is not incorporated in the mechanism. Therefore, this seems to be targeted at a single administrative domain.
5. Shared log based on network attached flash storage is emerging as an alternative for consensus. This system is not compared against a system such as CORFU.
**Is the approach technically sound?**
As with all new proposals, this proposal comes with certain constraints on the kinds of systems it applies to. Although these assumptions are not stated explicitly, they restrict the problem space, to which this solution applies. Focusing on the specific requirements of the solution automatically defines this restricted problem space and the resulting solution makes for very interesting study. The goals of the solution are defined clearly and the resulting solution is developed so that it fits naturally for the problem space.

**Organisation and Presentation**
The presentation of this paper makes it very clear what the goals of the developing this approach are. Another aspect I liked very much about this paper is that it was compared with a single other approach, which made it easy to understand what trade-offs were made differently. Although some of the details were omitted, this allows the authors to focus on core design details. A tech report was referenced for more details of the system design itself for those interested. This struck me as a good approach when pressed for space.

**Further work**
As mentioned in an earlier section, this is a fairly new paper which proposes a new approach for consensus. Another approach proposed has been to use a flash based shared log for consensus, which has been made possible due to the characteristics of flash memory. One obvious direction would be to look at the working of RAFT vis-a-vis that of using a shared log for consensus and the types of systems for which it is advantageous to use one or the other.