Efficient DAG-Based Consensus

Alberto Sonnino
Mysten Labs, London, UK

Abstract

This talk shows how to build high-performant Byzantine fault-tolerant (BFT) quorum-based consensus cores. The talks starts by challenging the common misconception that the overall communication complexity of the protocol is the key factor determining performance. We instead argue that the bottleneck of many state-of-the-art consensus protocols is their sequential use of the machine's resources (network, storage, CPU), and that data dissemination is the most resource-intensive task.

In light of the above considerations, the first insight to build performant BFT-based consensus cores is to separate the task of reliable transaction dissemination from transaction ordering. We show how to design a new DAG-based mempool protocol, called Narwhal, specialising in high-throughput reliable dissemination and storage of causal histories of transactions. Narwhal tolerates an asynchronous network and maintains high performance despite failures. It is designed to easily scale-out using multiple workers at each validator to concurrently use the machine's resources (network, storage, CPU), and demonstrates that there is no foreseeable limit to the throughput we can achieve. We then present two ways to leverage Narwhal to achieve consensus. We first (i) present Tusk, a zero-message overhead asynchronous consensus protocol designed to work with Narwhal. Tusk achieves an unprecedented 160,000 tx/s with about 3 seconds latency in a geo-replicated environment. We then (ii) show how any partially-synchronous consensus, such as HotStuff (PODC 19), can be composed with Narwhal to drastically improve its performance. HotStuff running over Narwhal sees its throughput increase from about 2,000 tx/s to over 130,000 tx/s without noticeable latency increase.

The talk concludes by illustrating how to properly evaluate performance of BFT-based consensus cores. It highlights the most common mistakes seen in the literature, such as benchmarks with empty transactions (empty load), performance approximation based on LAN-only benchmarks, and using a single burst of input transactions. We then show how to analyse benchmark results using latency-throughput graphs (L-graphs) and SLA-based throughput graphs.

Author Bio. I am a system researcher at Mysten Labs, based in London (UK). I previously was a research scientist at Facebook (now called Meta) in the blockchain and cryptography team. Before joining Facebook, I co-founded chainspace.io which built a scalable smart contract platform; the team was then acquired by Facebook. My research interests are in systems security and privacy engineering. My main areas of research include distributed systems, blockchains, and privacy enhancing technologies. I have a special interest in cryptography, and I spend most of my time designing, implementing and evaluating high-performance distributed systems.

2012 ACM Subject Classification Security and privacy → Distributed systems security

Keywords and phrases Consensus protocol, Byzantine Fault Tolerant

Digital Object Identifier 10.4230/OASIcs.FAB.2022.4

Category Invited Talk

Funding The majority of this work has been done when the authors were part of the Novi team at Facebook.

Acknowledgements This talk is based on the paper "Narwhal and Tusk: A DAG-based Mempool and Efficient BFT Consensus" (EuroSys 22) authored by George Danezis, Lefteris Kokoris-Kogias, Alberto Sonnino, and Alexander Spiegelman.