

Improving the Latency and Throughput of ZooKeeper Atomic Broadcast

Ibrahim EL-Sanosi^{1,2} and Paul Ezhilchelvan²

1 Faculty of Information Technology, Sebha University, Sebha, Libya
i.elsanosi@sebhau.edu.ly

2 School of Computing, Newcastle University, Newcastle Upon Tyne, UK
{i.s.el-sanosi,paul.ezhilchelvan}@ncl.ac.uk

Abstract

ZooKeeper is a crash-tolerant system that offers fundamental services to Internet-scale applications, thereby reducing the development and hosting of the latter. It consists of $N \geq 3$ servers that form a replicated state machine. Maintaining these replicas in a mutually consistent state requires executing an Atomic Broadcast Protocol, *Zab*, so that concurrent requests for state changes are serialised identically at all replicas before being acted upon. Thus, ZooKeeper performance for update operations is determined by *Zab* performance. We contribute by presenting two easy-to-implement *Zab* variants, called *ZabAC* and *ZabAA*. They are designed to offer small atomic-broadcast latencies and to reduce the processing load on the primary node that plays a leading role in *Zab*. The former improves ZooKeeper performance and the latter enables ZooKeeper to face more challenging load conditions.

1998 ACM Subject Classification D.2.8 Performance measures, D.4.7 Distributed systems

Keywords and phrases Atomic Broadcast, Server Replication, Protocol Latency, Throughput

Digital Object Identifier 10.4230/OASICS.ICCSW.2017.03



© Ibrahim EL-Sanosi and Paul Ezhilchelvan;
licensed under Creative Commons License CC-BY
2017 Imperial College Computing Student Workshop (ICCSW 2017).
Editors: Fergus Leahy and Juliana Franco; Article No. 03; pp. 03:1–03:1
Open Access Series in Informatics



OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

ICCSW17