

Challenges to Achieving High Availability at Scale

Wolfram Schulte

Facebook, Menlo Park, CA, USA
wolfram.schulte@outlook.com

Abstract

Facebook is a social network that connects more than 1.8 billion people. To serve these many users requires infrastructure which is composed of thousands of interdependent systems that span geographically distributed data centers. But what is the guiding principle for building and operating these systems?

For Facebook's infrastructure teams the answer is: Systems must always be available and never lose data. This talk will explore this quest. We will focus on three aspects.

Availability and consistency. What form of consistency do Facebook's systems guarantee? Strong consistency makes understanding easy but has latency penalties, weak consistency is fast but difficult to reason for developers and users. We describe our usage of eventual consistency and delve into how Facebook constructs its caching and replicated storage systems to minimize the duration for achieving consistency. We share empirical data that measures the effectiveness of our design.

Availability and correctness. With network partitions, relaxed forms of consistency, and software bugs, how do we guarantee a consistent state? We present two systems to find and repair structural errors in Facebook's social graph, one batch and one real-time.

Availability and scale. Sharding is one of the standard answers to operate at scale. But how can we develop one system that can shard storage as well as compute? We will introduce a new Sharding-as-a-Service component. We will show and evaluate how its design and service policies control for latency, failure tolerance and operationally efficiency.

1998 ACM Subject Classification Computer; C 1.4 Distributed Architectures; C.2.4 Distributed Systems; C.4 Fault Tolerance, Reliability, Availability and Serviceability; D 1.3 Distributed Programming; D 4.7 Distributed Systems; E 1 Distributed Data Structures

Keywords and phrases Distributed Systems, Availability, Reliability, Fault Tolerance, Consistency, Scalability, Replication, Sharding, Caching

Digital Object Identifier 10.4230/LIPIcs.ECOOP.2017.1

Category Invited Talk



© Wolfram Schulte;

licensed under Creative Commons License CC-BY

31st European Conference on Object-Oriented Programming (ECOOP 2017).

Editor: Peter Müller; Article No. 1; pp. 1:1–1:1

Leibniz International Proceedings in Informatics



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