

ZooKeeper: Because Coordinating Distributed Systems is a Zoo

ZooKeeper is a high-performance coordination service for distributed applications. It exposes common services - such as naming, configuration management, synchronization, and group services - in a simple interface so you don't have to write them from scratch. You can use it off-the-shelf to implement consensus, group management, leader election, and presence protocols. And you can build on it for your own, specific needs.

The following documents describe concepts and procedures to get you started using ZooKeeper. If you have more questions, please ask the [mailing list](#) or browse the archives.

- **ZooKeeper Overview**

Technical Overview Documents for Client Developers, Administrators, and Contributors

- [Overview](#) - a bird's eye view of ZooKeeper, including design concepts and architecture
- [Getting Started](#) - a tutorial-style guide for developers to install, run, and program to ZooKeeper

- **Developers**

Documents for Developers using the ZooKeeper Client API

- [API Docs](#) - the technical reference to ZooKeeper Client APIs
- [Programmer's Guide](#) - a client application developer's guide to ZooKeeper
- [ZooKeeper Java Example](#) - a simple Zookeeper client application, written in Java
- [Barrier and Queue Tutorial](#) - sample implementations of barriers and queues
- [ZooKeeper Recipes](#) - higher level solutions to common problems in distributed applications

- **Administrators & Operators**

Documents for Administrators and Operations Engineers of ZooKeeper Deployments

- [Administrator's Guide](#) - a guide for system administrators and anyone else who might deploy Zookeeper

- **Contributors**

Documents for Developers Contributing to the ZooKeeper Open Source Project

- [ZooKeeper Internals](#) - assorted topics on the inner workings of ZooKeeper

- **Informal Documentation**

Miscellaneous, Informal ZooKeeper Documentation

- [Wiki](#)
- [FAQ](#)