

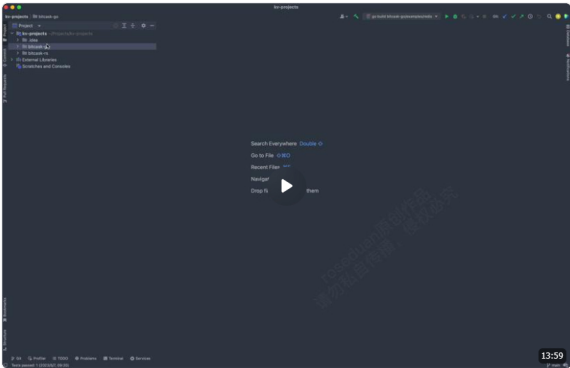
<<
18 从零实现 KV 存储—基准测试
概述
扩展内容—YCSB 测试
扩展内容—和其他存储引擎的对比

18 从零实现 KV 存储—基准测试

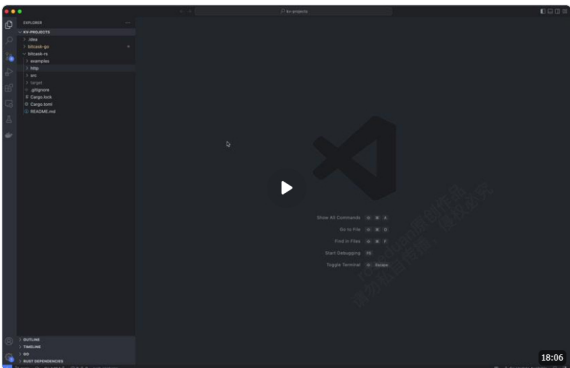
理论讲解



Go 编码部分



Rust 编码部分



概述

基准测试 (benchmark) 指的是对系统的性能进行量化测量, 对于存储引擎这样的偏底层系统, 基准测试非常重要, 因为它能够帮助开发者和使用者了解系统的性能指标, 查看系统的瓶颈, 并且针对性的去调优。

对于存储引擎以及其他的基础框架, 基准测试通常会关注以下几个方面:

1. 吞吐量: 指单位时间内能够处理的请求数量。对于存储引擎来说, 吞吐量的大小会直接影响到系统的整体性能。
2. 响应时间: 指完成一个请求所需的时间。响应时间长短直接影响用户体验, 因此这也是存储引擎性能评估的一个关键指标。
3. 并发度: 指系统能够同时处理的请求数量, 并发度的大小会影响到系统的稳定性和吞吐量。

通过对这些指标进行量化测量, 我们可以对存储引擎的性能进行全面的评估, 并且可以根据测试结果来改进实现, 从而提高系统的性能。

对于 Go 语言, 我们在课程中将会采用标准库中自带的 testing 框架进行基准测试。

对于 Rust 语言, 其实标准库也有对应的 benchmark 框架, 但是无法在 Rust 的 stable 版本中进行使用, 所以我们可以使用应用更加广泛的 <https://github.com/bheisler/criterion.rs> 这个库。

这一节我将会为我们的存储引擎的 Put、Get、Delete 这三个方法添加基准测试, 在这基础之上, 你可以去添加其他方法的测试, 逻辑都是类似的。

扩展内容—YCSB 测试

YCSB 是一个用于测试和评估分布式存储系统性能的基准测试框架。它的全称是 Yahoo! Cloud Serving Benchmark, 由 Yahoo! Research 团队开发并开源, 目前已经成为了业界标准之一。

YCSB 提供了一个通用的测试接口, 可以与多种不同类型的存储系统 (例如关系型数据库、NoSQL 数据库、分布式文件系统等等) 进行集成。

<https://github.com/pingcap/go-ycsb>

扩展内容—和其他存储引擎的对比

Go

Badger

Pebble

Bolt

Rust

Sled

一些关于基准测试的参考资料：

<https://geektutu.com/post/hpg-benchmark.html>

<https://course.rs/test/benchmark.html>

<https://github.com/smallnest/kvbench>



1 人点赞



输入评论

