

从所有教程的词条中查询...

首页 > 慕课教程 > 物联网/嵌入式工程师 > 3.1 线程池原理与框架设计

全部开发者教程

12.8 多路复用 io-epoll(三)-原理分析

第13周 Linux 项目实战-实现并发服务器模型与企业笔试

1.1 项目框架介绍

2.1 环形队列设计(一)-基本框架与共享内存模块设计

2.2 环形队列设计(二)-环形队列初始化与销毁

2.3 环形队列设计(三)-环形队列数据读写实现

3.1 线程池原理与框架设计

3.2 线程池的数据结构设计

3.3 线程池初始化

3.4 线程池添加任务

3.5 线程执行函数

3.6 线程池的销毁

3.7 线程池的测试

4.1 项目模块整合与数据结构设计

4.2 消息订阅功能实现(一)-客户端初始化与订阅数据发送

4.3 消息订阅功能实现(二)-服务器接收与数据分发

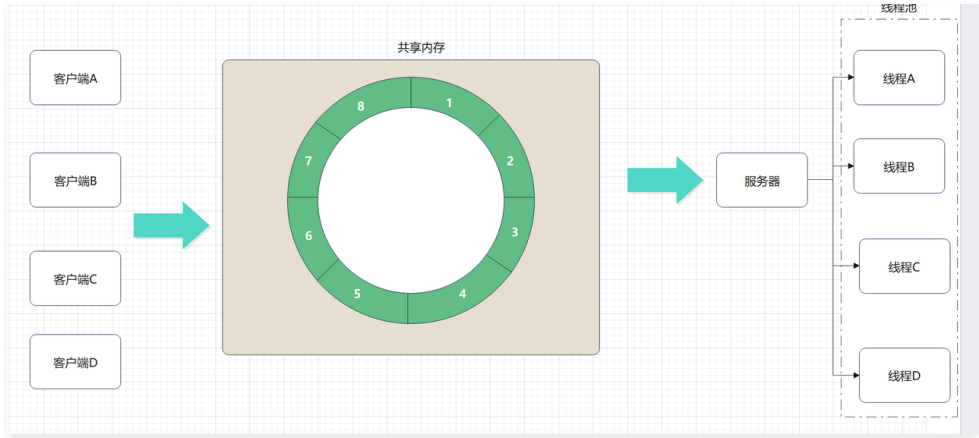


大白老师 · 更新于 2022-12-09

上一节 2.3 环形队列设计... 3.2 线程池的数据结构... 下一节

一、线程池的意义

- 问题 1：在多个客户端向服务器发送请求，如何提高效率？
- 解决方案：
 - 使用环形队列缓存请求,在服务器处理一个请求后，可以直接从队列中获取下一个请求，无需等待客户端发送
 - 使用多线程并发，将请求的处理分配给线程来完成，处理完成后则退出
- 问题 2: 在处理请求的过程中，需要创建线程与销毁线程，会导致资源的消耗，如何解决？
- 解决方案：
 - 使用线程池
 - 可以创建多个线程，在有任务的时候，则可以执行任务，在没有任务时，则进入睡眠状态
 - 减少创建与销毁线程的消耗



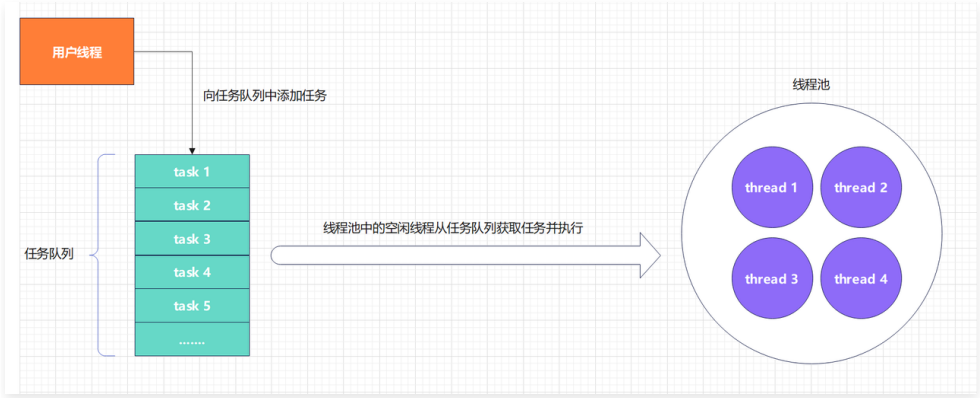
二、线程池的基本框架

- 线程池的基本框架:

意见反馈

收藏教程

标记书签



- step 1 : 线程池在初始化时会创建多个线程, 在没有任务时, 会进入睡眠状态
- step 2 : 用户线程可以向任务队列添加任务, 并通过条件变量唤醒空闲线程, 执行任务
- step 3 : 当任务队列已满时, 则用户线程会进入睡眠状态, 在线程池中的线程处理任务之后, 则会唤醒用户线程
- 注意:
 - 这里用户线程与线程池中的线程是生产者与消费者模型
 - 用户线程负责生产任务, 线程池中的线程负责消费任务(处理任务)
 -

2.3 环形队列设计(三)-环形队列数据读写实现 ◀ 上一节 下一节 ▶ 3.2 线程池的数据结构设计

✎ 我要提出意见反馈