

4.13 消息发布功能实现 (七)- 服务器转发发布消息_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 4.13 消息发布功能实现 (七)- 服务器转发发布消息涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

- 当服务器接收到客户端发布的消息, 就需要将消息转发给所有已经订阅的进程, 具体逻辑如下:
 - step 1: 查询 hashmap , 并获取所有订阅进程的 pid
 - step 2: 将订阅进程的 pid 作为消息类型, 在封装消息内容, 写入到消息队列
- step 1: 将 msg.h msg.c 拷贝服务器的工程目录中

```
ben@ubuntu:~/class/week13/workspace/project/version1/server$ ls
hashmap.c  main.c  msg.c  sem.c  server.c  shm.c  shmfifo.h  thread_pool.c
hashmap.h  Makefile  msg.h  sem.h  server.h  shmfifo.c  shm.h  thread_pool.h
```

- step 2: 修改 Makefile , 添加 msg.o

```
OBJS := sem.o shmfifo.o shm.o server.o main.o thread_pool.o hashmap.o msg.o

server:$(OBJS)
@gcc $^ -o $@ -lpthread
@echo "Done."

%.o : %.c
@gcc -c $< -o $@

clean:
@rm *.o server
@echo "Clean."
```

```
#include "shmfifo.h"
#include "server.h"
#include "thread_pool.h"
#include "hashmap.h"
#include "msg.h"
static shm_fifo_t *g_shmfifo = NULL;
static tpool_t *g_tpool = NULL;
static hashtable_t **g_hashtable = NULL;
static int g_msgid = 0;

void server_init()
{
    g_shmfifo = shmfifo_init(BLOCK_CNT, sizeof(packet_t));
    g_tpool = thread_pool_init(THREAD_CNT, TASK_CNT);
    g_hashtable = create_hashtable();
    g_msgid = msg_create();
}
```

- step 4: 服务器转发消息给所有订阅的客户端

```
void do_publish(char *topic, char *content)
{
    pid_t result[RESULT_SZ];
    int ret;
#ifdef DEBUG
    printf("do publish.\n");
#endif

    ret = search_hash_table(g_hashtable, topic, result);
    if (ret != 0){
```

```
#ifdef DEBUG
    printf("ret = %d\n",ret);
#endif
    for (int i = 0;i < ret;i++){
        msg_send(g_msgid,result[i],content) ;
    }
}
```

- 1. 在线程池拓展新的功能，能够根据任务情况动态添加线程或者减少线程
- 2. 在同一个进程订阅主题消息，hashmap 会重复，请 解决这个问题
- 3.hashmap 是属于临界资源，但没有保护，请在项目中添加保护

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

