message queue in C: implementing 2 way comm

I am a student and a begineer in C. I want to implement 2 way communication using message queue in C linux. Do I need two queues or only one to get this done?

Also I would like to know can I send data(shown in code) to another process or i need to declare it as a character array.

```
typedef struct msg1
{
    int mlen;
    char *data;
}M1;

typedef struct msgbuf
{
    long    mtype;
    M1      *m;
} message_buf;
```

Thanks in advance :)

c linux pointers ipc message-queue

asked Mar 26 '14 at 12:50

user3433848
52 1 6

If you want to send messages between processes, read more about IPC, and use something like POSIX message queues, because you can't do it by just sending e.g. pointers between processes (as each process has its own private memory map). – Some programmer dude Mar 26 '14 at 12:56

And yes, for two-way communication you need two queues. – Some programmer dude Mar 26 '14 at 12:56

you will need multiple message queues. Trying to use a single message queue for two-way communication between multiple processes would turn into a complex situation – Jayesh Mar 26 '14 at 12:57

thanks a lot @JoachimPileborg . – user3433848 Mar 26 '14 at 13:00

@Jayesh I just wanted to know if its possible? Is it? – user3433848 Mar 26 '14 at 13:01

1 Answer

```
Also I would like to know can I send data(shown in code) to another process or i need to
declare it as a character array
```

yes you can send data to another process

like

```
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define MAXSIZE     128

void die(char *s)
{
  perror(s);
```

```
  exit(1);
}

struct msgbuf
{
    long    mtype;
    char    mtext[MAXSIZE];
};

main()
{
    int msqid;
    int msgflg = IPC_CREAT | 0666;
    key_t key;
    struct msgbuf sbuf;
    size_t buflen;

    key = 1234;

    if ((msqid = msgget(key, msgflg )) < 0)   //Get the message queue ID for the given key
      die("msgget");

    //Message Type
    sbuf.mtype = 1;

    printf("Enter a message to add to message queue : ");
    scanf("%[^\n]",sbuf.mtext);
    getchar();

    buflen = strlen(sbuf.mtext) + 1 ;

    if (msgsnd(msqid, &sbuf, buflen, IPC_NOWAIT) < 0)
    {
        printf ("%d, %d, %s, %d\n", msqid, sbuf.mtype, sbuf.mtext, buflen);
        die("msgsnd");
    }

    else
        printf("Message Sent\n");

    exit(0);
}


//IPC_msgq_rcv.c

#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>
#include <stdio.h>
#include <stdlib.h>
#define MAXSIZE     128

void die(char *s)
{
  perror(s);
  exit(1);
}

typedef struct msgbuf
{
    long    mtype;
    char    mtext[MAXSIZE];
} ;


main()
{
    int msqid;
    key_t key;
    struct msgbuf rcvbuffer;

    key = 1234;

    if ((msqid = msgget(key, 0666)) < 0)
      die("msgget()");


     //Receive an answer of message type 1.
    if (msgrcv(msqid, &rcvbuffer, MAXSIZE, 1, 0) < 0)
      die("msgrcv");

    printf("%s\n", rcvbuffer.mtext);
    exit(0);
}
```

If you know about message queue, then Message Queue is used for inter-process communication.

Also for two-way communication between multiple processes you need multiple message queue

edited Mar 26 '14 at 13:09

answered Mar 26 '14 at 13:03

Jayesh
9,265 7 22 48

two way comm between given two processes only, requires how many queues atleast?? – user3433848 Mar 26 '14 at 13:15

two message queue required – Jayesh Mar 26 '14 at 13:18

Thanks a lot sir :) – user3433848 Mar 26 '14 at 13:19

@user3433848 you are welcome – Jayesh Mar 26 '14 at 13:20

@user3433848, you can do two-way communication with a sysv MQ but it requires a level of familiarity with queues you have yet to attain and, even under the best of circumstances, is a questionable practice unless you are really forced into it by circumstances. Two queues are easier all the way around. – Duck Mar 26 '14 at 16:01