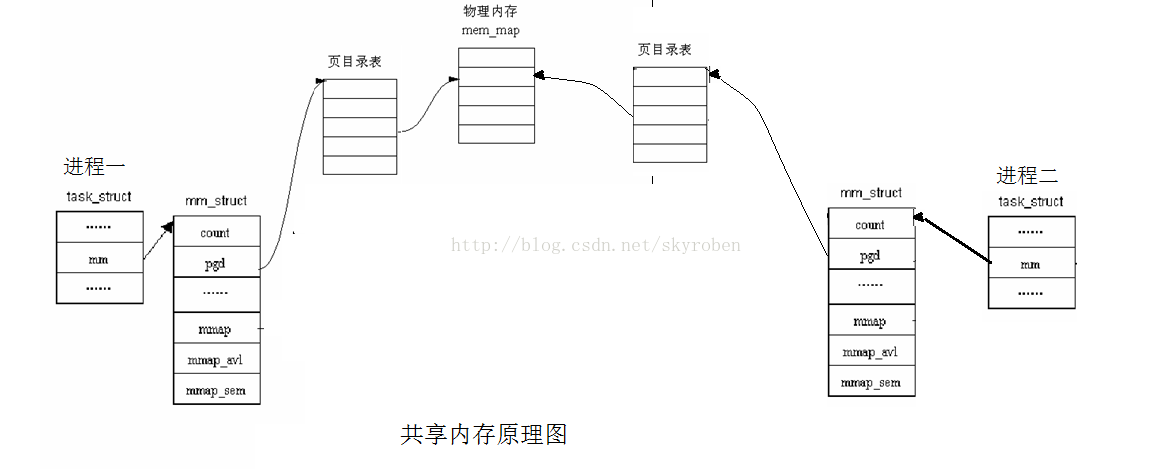
**共享内存**

# 定义

共享内存就是允许两个或多个进程共享一定的存储区。就如同 malloc() 函数向不同进程返回了指向同一个物理内存区域的指针。当一个进程改变了这块地址中的内容的时候，其它进程都会察觉到这个更改。因为数据不需要在客户机和服务器端之间复制，数据直接写到内存，不用若干次数据拷贝，所以这是最快的一种IPC。



# 常用函数

## **创建共享内存**

**int shmget(key\_t key, size\_t size, int shmflg);**

key：是长整型（唯一非零），系统建立IPC通讯 （ 消息队列、 信号量和 共享内存） 时必须指定一个ID值。通常情况下，该id值通过ftok函数得到，由内核变成标识符，要想让两个进程看到同一个信号集，只需设置key值不变就可以。

size：指定共享内存的大小，它的值一般为一页大小的整数倍（未到一页，操作系统向上对齐到一页，但是用户实际能使用只有自己所申请的大小）。

shmflg：是一组标志，创建一个新的共享内存，将shmflg 设置了IPC\_CREAT标志后，共享内存存在就打开。而IPC\_CREAT | IPC\_EXCL则可以创建一个新的，唯一的共享内存，如果共享内存已存在，返回一个错误。一般我们会还或上一个文件权限

## 操作共享内存

**int shmctl(int shm\_id, int cmd, struct shmid\_ds \*buf);**

shm\_id：是shmget函数返回的共享内存标识符。

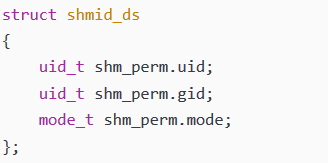
cmd：是要采取的操作，它可以取下面的三个值 ：

IPC\_STAT：把shmid\_ds结构中的数据设置为共享内存的当前关联值，即用共享内存 的当前关联值覆盖shmid\_ds的值。

IPC\_SET：如果进程有足够的权限，就把共享内存的当前关联值设置为shmid\_ds结 构中给出的值

IPC\_RMID：删除共享内存段

buf：是一个结构指针，它指向共享内存模式和访问权限的结构。 shmid\_ds结构至少包括以下成员



## 挂接操作

创建共享存储段之后，将进程连接到它的地址空间

**void \*shmat(int shm\_id, const void \*shm\_addr, int shmflg);**

shm\_id：是由shmget函数返回的共享内存标识。

shm\_addr：指定共享内存连接到当前进程中的地址位置，通常为空，表示让系统来选择共享内存的地址。

shm\_flg：是一组标志位，通常为0

## 分离操作

**int shmdt(const void \*shmaddr);**

addr：是以前调用shmat时的返回值

该操作不从系统中删除标识符和其数据结构，要显示调用shmctl(带命令IPC\_RMID)才能删除它