算法getblk()

//Entrée: 文件系统号 Numéro de système de fichiers

//        块号      Numéro de bloc

//Sorti:  现在能被磁盘块使用的上了锁的缓冲区 tampons verrouillé qui peut être utilisé pour le bloc

{

    while(tampons non trouvé) //没有找到缓冲区

    {

        if(bloc dans la file à adressage calculé) //如果块在散列队列中 第一种情况

        {

            if (bloc occupé) //块忙 第二种情况

            {

                sleep(événement: le tampons est libre) //等候缓冲区变为空闲这个事件

            }

            marquer tampon occupé; //为该缓冲区标记“忙”

            extraire le tampon de la liste des tampons libres; //从空闲表上摘下缓冲区

            return(tampon);

        }

        else // 块不在散列队里 Le tampon n'est pas dans la file à adressage calculé

        {

            if(pas de tampon dans la liste des tampons libres) //空闲表中没有缓冲区 第五种情况

            {

                sleep(événement: le tampon devient libre); //等待有缓冲区变为空闲事件

            }

            extraire le tampon de la liste des tampons libres; //从空闲表上摘下缓冲区

            if(tampon est marqué pour écriture différée) //如果缓冲区被标记为“延迟写” 第四种情况

            {

                écriture asychrone du tampon sur le disque; //把缓冲区异步的写到磁盘

            }else{ 第三种情况

                extraire le tampon de l'ancienne file ;  //从旧散列队列中摘下缓冲区

                mettre le tampon de la nouvelle file;  //把缓冲区投入新的散列队列中

            }

            return(tampon);

        }

    }

}

getblk中内核把一个缓冲区分配给磁盘块时可能出现的五种典型情况：

* 内核发现该块在散列队列中，并且它的缓冲区是空闲的

Le bloc est dans la file à adressage calculé et dans la liste des tampons libres.

* Le bloc a été utilisé récement et est disponible.
* 内核发现该块在散列队列中，但他的缓冲区被标记为"忙"

Le bloc est dans la file à adressage calculé mais pas dans la liste des tampons libres.

* Le bloc est en-cours d'utilisation =>attente courte
* 内核在散列队列中找不到该块，因此从空闲表中分配一个缓冲区

Le bloc n'est pas dans la file à adressage calculé.

* Attribution du tampon libre et lecture disque => petite perte du temps
* 内核在散列队列中找不到该块，在空闲表中找到一个被标记为"延迟写"的缓冲区，内核必须把"延迟写"缓冲区的内容异步的写到磁盘上，并分配一个新的缓冲区.

Le bloc n'est pas dans la file à adressage calculé et le premier tampon libre est en "ecriture différée".

* Ecriture + attribution du tampon suivant + lecture disque
* 内核在散列队列中找不到该块，并且没有缓冲区在空闲表中

Le bloc n'est pas dans la file à adressage calculé et il n'y a plus de tampons libres.

* Attente + attribution du tampon suivant + lecture disque

算法 brelse()

//Entrée: 等待释放的缓冲区 tampon vérrouillé

//Sorti:  无 néant

{

    reveiller tous les processus en attente d'un tampon libre; //唤醒正在等待“无论哪个缓冲区变为空闲”这一事件的所有进程

    reveiller tous les processus en attente que ce tampon soit libre; //唤醒正在等待“这个缓冲区变为空闲”这一事件的所有进程

    reler le niveau d'exécution du processus pour inhiber les IT //提高处理机执行级以封锁中断

    if(contenu du tampon valide et tampon non ancien) //缓冲区内容有效且为非"旧"

        mettre le tampon en queue de la liste des tampons libres; //将缓冲区置入空闲表尾部

    else

        mettre le tampon en tête de la liste des tampons libres; //将缓冲区置入空闲表头部

    rabaisser le niveau d'exécution du processus pour autoriser les IT;//降低处理机执行级以允许中断

    déverrouiller le tampon;

}

算法bread()

//Entrée: 文件系统块号 numéro de bloc dans la système de fichiers

//Sorti:  含有数据的缓冲区 tampon contenant les données

{

    obtenir un tampon pour le bloc (algorithme getblk()); //得到该块的缓冲区

    if(données du tampon valides) //缓冲区数据有效

    {

        return tampon;

    }

    initialiser une lecture disque ;//启动磁盘读

    sleep(événement: fin de la lecture disque) //等待“读盘完成”事件

    return tampon;

}

算法bwrite()

//Entrée: 缓冲区 tampon

//Sorti:  无 néant

{

    initialiser une écriture disque ; //启动磁盘写

    if(E/S synchrone) //I/O同步

    {

        sleep(événement: fin de lE/S); //等待I/O完成事件

        libérer tampon (algo brelse()); //释放缓冲区

    }

    else if(tampon marqué pour une écriture différée) //缓冲区标记着延迟写

        marquer tampon à mettre en tête de la liste des tampons libre; //为缓冲区做标记以放到空闲表头部

}

算法breada()

//Entrée: 立即文件系统块号 numéro de bloc du SF pour lecture immédiate

//        异步读取文件系统块号 numéro de bloc du SF pour lecture asynchrone

//Sorti:  含有可立即读的数据的缓冲区 tampon contenant les données en lecture immédiate

{

    if(premier bloc absent du cache) //第一块不在告诉缓冲中

    {

        obtenir un tampon pour premier bloc;  //getblk() 为第一块获得缓冲区

        if(données du tampon non valides)   //缓冲区数据无效

            initiliser une lecture disque;  //启动磁盘读

    }

    if(second bloc absent du cache) //第二块不在告诉缓冲中

    {

        obtenir un tampon pour second bloc;  //getblk() 为第二块获得缓冲区

        if(données du tampon valides) //缓冲区数据有效

            libérer le tampon; //brelse() 释放缓冲区

        else

            initiliser une lecture disque; //启动磁盘读

    }

    if(premier bloc était originellement dans le cache) //第一块本来在告诉缓冲中

    {

        Accéder au premier bloc; //bread() 读第一块

        return tampon;

    }

    sleep(événement: le premier tampon contient des données valides);//第一个缓冲区包含有效数据事件

    return tampon;

}