**Linux缓冲机制探究**

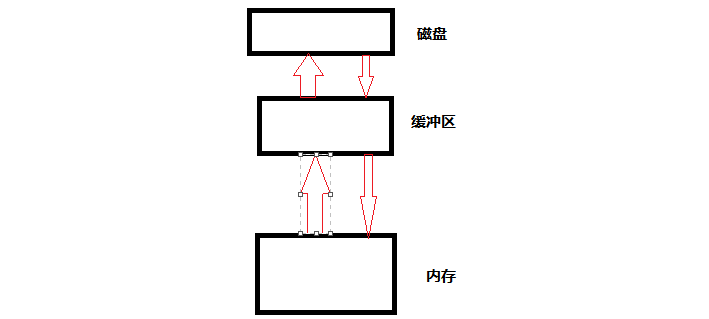
作者：chw

1. 简介

缓冲区又称为缓存，它是内存空间的一部分。也就是说，在内存空间中预留了一定的存储空间，这些存储空间用来缓冲输入或输出的数据，这部分预留的空间就叫做缓冲区。

缓冲区根据其对应的是输入设备还是输出设备，分为输入缓冲区和输出缓冲区。

思考：为什么要引入缓冲区呢？



比如我们从磁盘里取信息，我们先把读出的数据放在缓冲区，计算机再直接从缓冲区中取数据，等缓冲区的数据取完后再去磁盘中读取，这样就可以减少磁盘的读写次数，再加上计算机对缓冲区的操作大大快于对磁盘的操作，故应用缓冲区可大大提高计算机的运行速度。

（例如，我们使用打印机打印文档，由于打印机的打印速度相对较慢，我们先把文档输出到打印机相应的缓冲区，打印机再自行逐步打印，这时我们的CPU可以处理别的事情。）

总结：缓冲区就是一块内存区，它用在输入输出设备和CPU之间，用来缓存数据。它使得低速的输入输出设备和高速的CPU能够协调工作，避免低速的输入输出设备占用CPU，解放出CPU，使其能够高效率工作。

二、缓冲区的分类

<1>全缓存[4096byte]

特点：缓存满的时候会刷新缓存，此时进行实际的IO写操作。（写到磁盘中）。一般我们使用stdio库对文件的操作均是全缓冲。（注：刷新就是将缓存写到硬盘上的意思。）

刷新条件：(1)缓存满的时候会刷新

(2)程序正常结束的时候会刷新 [调用exit或从main函数返回]

(3)调用fflush（）函数会强制刷新缓存 [fflush为刷新缓存的函数]

<2>行缓存[1024byte]

特点：输入输出时遇'\n'会刷新。stdio库对与终端相关联的文件使用行缓冲,

典型的有标准输入文件和标准输出文件。(stdin,stdout)

刷新条件：(1)缓存满的时候会刷新

(2)程序正常结束的时候会刷新 [调用exit或从main函数返回]

(3)调用fflush（）函数会强制刷新缓存 [fflush为刷新缓存的函数]

(4)遇到’\n’的时候会刷新缓存。 [独有的属性]

<3>不缓存[0byte]

特点：没有缓存，每次都是直接调用系统接口！

典型的有标准出错。（stderr）

标准IO库的设计原则：

默认情况下，打开的文件都是全缓存类型。当于终端相关联的时候，就是行缓存。

例如：stdin 、stdout是行缓存。

Stderr是不缓存。