

第八章 磁盘存储器的管理



第八章 磁盘存储器的管理

8.1 外存组织方式

8.2 文件存储空间的管理

8.3 提高磁盘I/O速度的途径

8.4 提高磁盘可靠性的技术



第八章 磁盘存储器的管理

8.1 外存组织方式

8.2 文件存储空间的管理

8.3 提高磁盘I/O速度的途径

8.4 提高磁盘可靠性的技术



8.1 外存组织方式

由于磁盘具有可直接访问的特性，故当利用磁盘来存放文件时，具有很大的灵活性。

在为文件分配外存空间时所要考虑的主要问题是：怎样才能有效的利用外存空间和如何提高对文件的访问速度。

目前常用的外存分配方法有：连续分配、链接分配和索引分配三种。

在一个系统通常只采用一种方法。

文件的物理结构和外存分配方法有关。在采用不同的分配方式时将形成不同的文件物理结构，如：

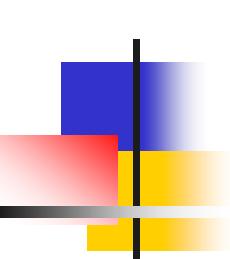
- 连续方式对应顺序结构；
- 链接分配方式形成链接式结构；
- 索引方式形成索引式结构。

1 连续组织分配

- 连续分配要求为每一个文件分配一组相邻的盘块。在采用该方式时，可把逻辑文件中的记录顺序的存储到邻接的各物理块中，这样所形成的文件结构成为顺序文件结构，此时的物理文件称为顺序文件。这种分配方式保证了逻辑文件中的记录顺序与存储器中文件占用盘块的顺序的一致性。
- 随着文件的建立与删除不断进行，将产生很多外存的碎片，利用紧凑方法也可消除碎片。

优点：顺序访问容易； 顺序访问速度快。

缺点：要求有连续的存储空间； 必须事先知道文件的长度。



2 链接组织方式

- 采用链接分配方式时，可通过在每个盘块上的链接指针，将同属于一个文件的多个离散的盘块链接成一个链表，把这样形成的文件称为链接文件。
- 链接方式又可分为**隐式链接**和**显式链接**两种形式。

隐式链接 p252

显式链接 p253

- 链式分配的特点
 - 由于链式分配是采取**离散分配方式**，消除了外部碎片；
 - 因为是根据文件的当前需要，为它分配必须的盘块，当**文件动态增长**时，可动态的再为它分配盘块，故而无需事先知道文件的大小。
 - 对文件的增、删、改也十分方便。

例、假定磁盘块大小为1K，对于1.2G的硬盘，其文件分配表FAT需占用多少存储空间？

答： (1) 当硬盘大小为1.2G， 盘块大小为1K时， 硬盘共有的盘块数为：

$$1.2\text{G} / 1\text{K} = 1.2\text{M} \text{ (个)}$$

由于 $1\text{M} < 1.2\text{M} < 2\text{M}$

所以，1.2M个盘块要用21位二进制表示。为方便FAT的存取，每个表目用32位2进制表示，即FAT的每个表目为4个字节。

FAT要占用的存储空间总数为：

$$4 * 1.2\text{M} = 4.8 \text{ MB}$$

3 索引组织方式

链接分配方式虽然解决了连续分配方式所存在的问题，但又出现了另外两个问题：

不能支持高效的直接存取。要对一个文件进行直接存取，需首先在FAT中顺序的查找许多盘块号。

FAT需占用较大的内存空间。当磁盘容量较大时，FAT可能要占用数MB以上的内存空间。这是令人难以忍受的。

索引分配方式示意图 见P257图8-6；图8-7

单级索引方式

多级索引方式

混合索引方式

索引分配方式的问题

可能要花费较多的外存空间。每当建立一个文件时，便须为之分配一个索引块，将分配给该文件的所有盘块号记录于其中。

文件的逻辑结构、物理结构 总结

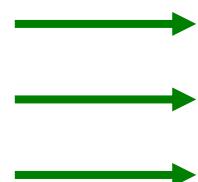
■ 文件的逻辑结构

- 1) 有结构的记录式文件（定长记录、不定长记录）
 - ① 顺序文件（串结构、顺序结构）
 - ② 索引文件
 - ③ 索引顺序文件
 - ④ 直接文件、哈希文件
- 2) 无结构的流式文件

■ 文件的物理结构

与外存的分配方式有关，例如：

连续分配



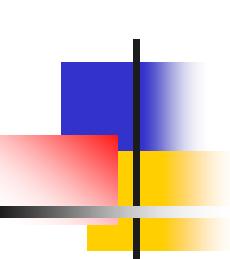
顺序文件

链接分配

链接文件

索引分配

索引文件（混合索引文件）



第八章 磁盘存储器的管理

8.1 外存组织方式

8.2 文件存储空间的管理

8.3 提高磁盘I/O速度的途径

8.4 提高磁盘可靠性的技术

8.5 数据一致性控制

8.2 文件存储空间的管理

文件管理要解决的重要问题之一是如何为**新创建**的文件分配存储空间。其解决方法与内存的分配情况有许多相似之处

连续分配方式

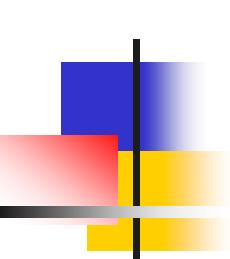
离散分配方式

存储空间的基本分配单位都是
磁盘块而非字节。

为了实现存储空间的分配，首先必须记住空闲存储空间的使用情况。

数据结构

分配和回收功能



文件存储
空间的管理
方法

空闲表法 和 空闲链表法

位示图法

成组链接法

重点
难点

8.2.1 空闲表法和空闲链表法

空闲表法

空闲表

分配与回收

属于连续分配方式。与内存的动态分配方式雷同，它为每个文件分配一块连续的存储空间。系统为外存上的所有空闲区建立一张空闲表，每个空闲区对应于一个空闲表项，再将所有空闲区按其起始盘块号递增的次序排列

序号	第一空闲盘块号	空闲盘块数
1	2	4
2	9	3
3	15	5
4	—	—

8.2.1 空闲表法和空闲链表法

空闲链表法

空闲盘块链

空闲盘区链

这是将磁盘上的所有**空闲盘区**拉成一条链。在每个盘区上除含有用于指示下一个**空闲盘区**的指针外，还应有能指明本盘区大小的信息。**分配盘区的方法与内存动态分区分配类似**，通常采用首次适应算法。在回收盘区时，同样也要将回收区与相邻的**空闲盘区**相合并。在采用首次适应算法时，为提高对**空闲盘区**的检索速度，可以采用显式链接方法，亦即，在内存中为**空闲盘区**建立一张链表。

8.2.2 位示图法

位示图

是利用二进制的一位来表示磁盘中一个盘块的使用情况。当其值为0时表示对应的盘块空闲，为1时表示已分配（Linux）。有的系统则相反。磁盘上的所有盘块都有一个二进制位与之对应，这样由所有盘块所对应的位构成一个集合，称为位示图。通常可用 $m*n$ 个位数来构成位示图，并使 $m*n$ 等于磁盘的总块数。

二维数组

Var map: array[1...m, 1...n] of bit



八一版

盘块的分配

顺序扫描位示图。以找到一个或一组其值为“0”的二进制位；

位示图



盘块的分配

顺序扫描位示图。以找到一个或一组其值为“0”的二进制位；

位示图



盘块的分配

将所找到的一个或一组二进制位转换成盘块号 $b = \text{列数}n * (i-1) + j;$

位示图



b=3;b=4;b=5

位示图

盘块的分配

修改位示图。Map[i,j]=1

位示图



将回收盘块的盘块号转换成位示图中的行号和列号。转换公式为：

i=(b-1)DIV n+1

j=(b-1)MOD n+1

位示图



修改位示图

Map[i,j]=0;

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

位示图



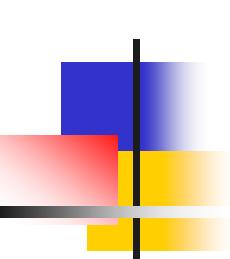
修改位示图

Map[i,j]=0;

位示图

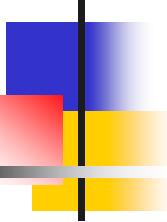


从位示图中很容易找到一个或一组相邻接的空闲盘块。此外，由于位示图很小，占用空间少，因而可将它保存在内存中，从而在每次进行盘区分配时，无需首先把磁盘分配表读入内存，从而省掉许多磁盘的启动操作。位示图常用于微型机和小型机中，如Linux



8.2.3 成组链接法

空闲表法和空闲链表法都不适用于大型文件系统，因为这会使空闲表或空闲链表太长。在UNIX系统中采用的是成组链接法，这是将上述两种方法结合而形成的一种空闲盘块管理方法，它兼备了上述方法的优点而克服了表太长的缺点。



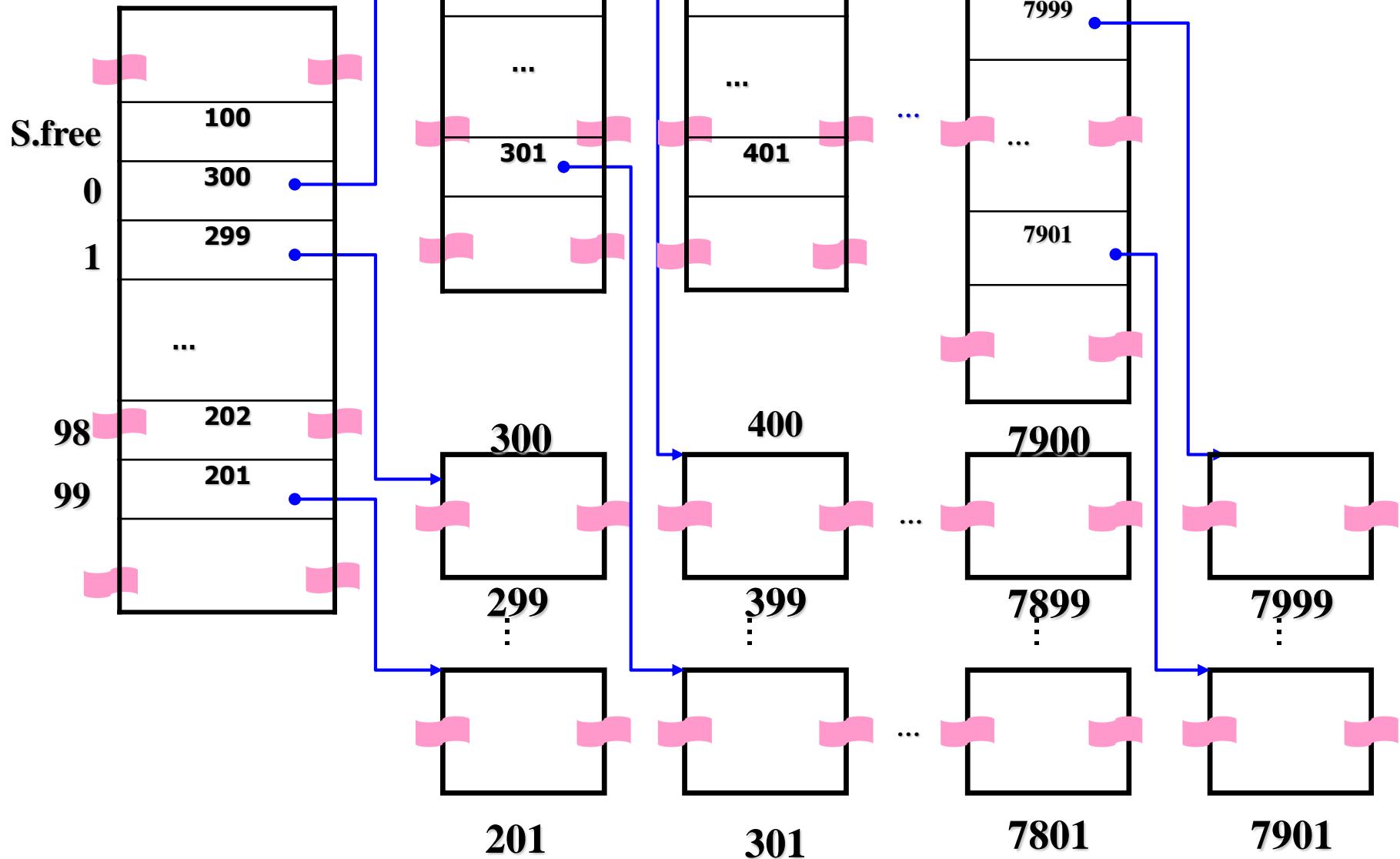
成组链接法

空闲盘块的组织

空闲盘块的分配与回收

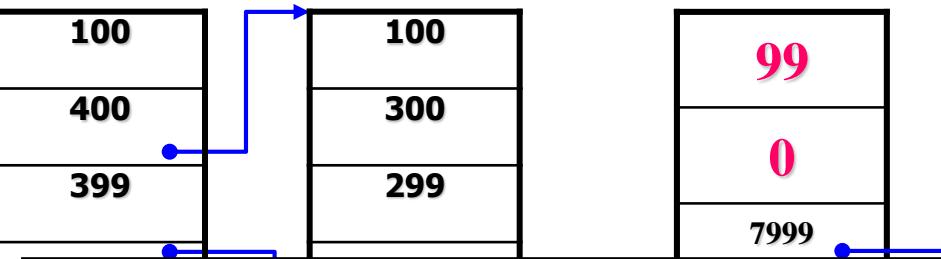
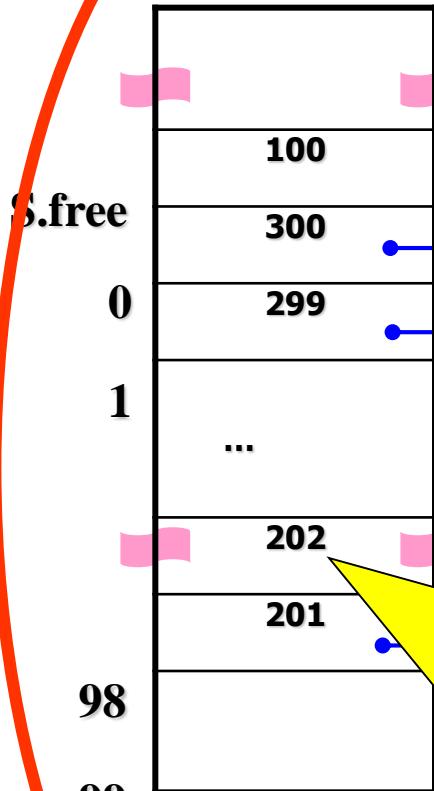
空闲盘块的组织

空闲盘块号栈



空闲盘块的组织

空闲盘块号栈



空闲盘块号栈。用来存放当前可用的一组空闲盘块的盘块号（最多含100个号），以及栈中尚有的空闲盘块号数N。N还兼具栈顶指针用。例如当N=100时，它指向S.free(99)。S.free(0)是栈底，栈满时栈顶为S.free(99)。由于栈是临界资源，每次只允许一个进程去访问，故系统为栈设置了一把锁。

信号量

201

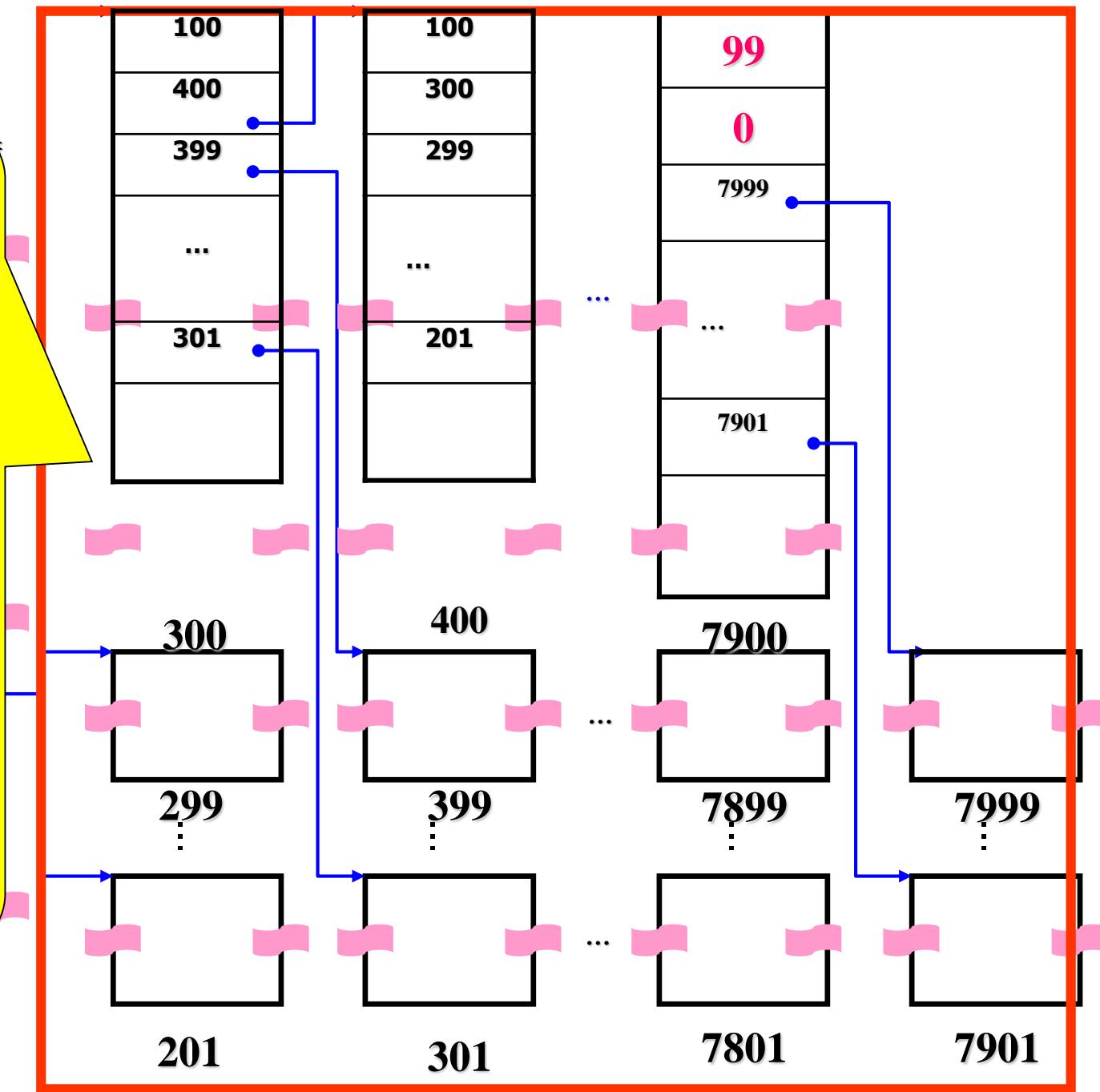
301

7801

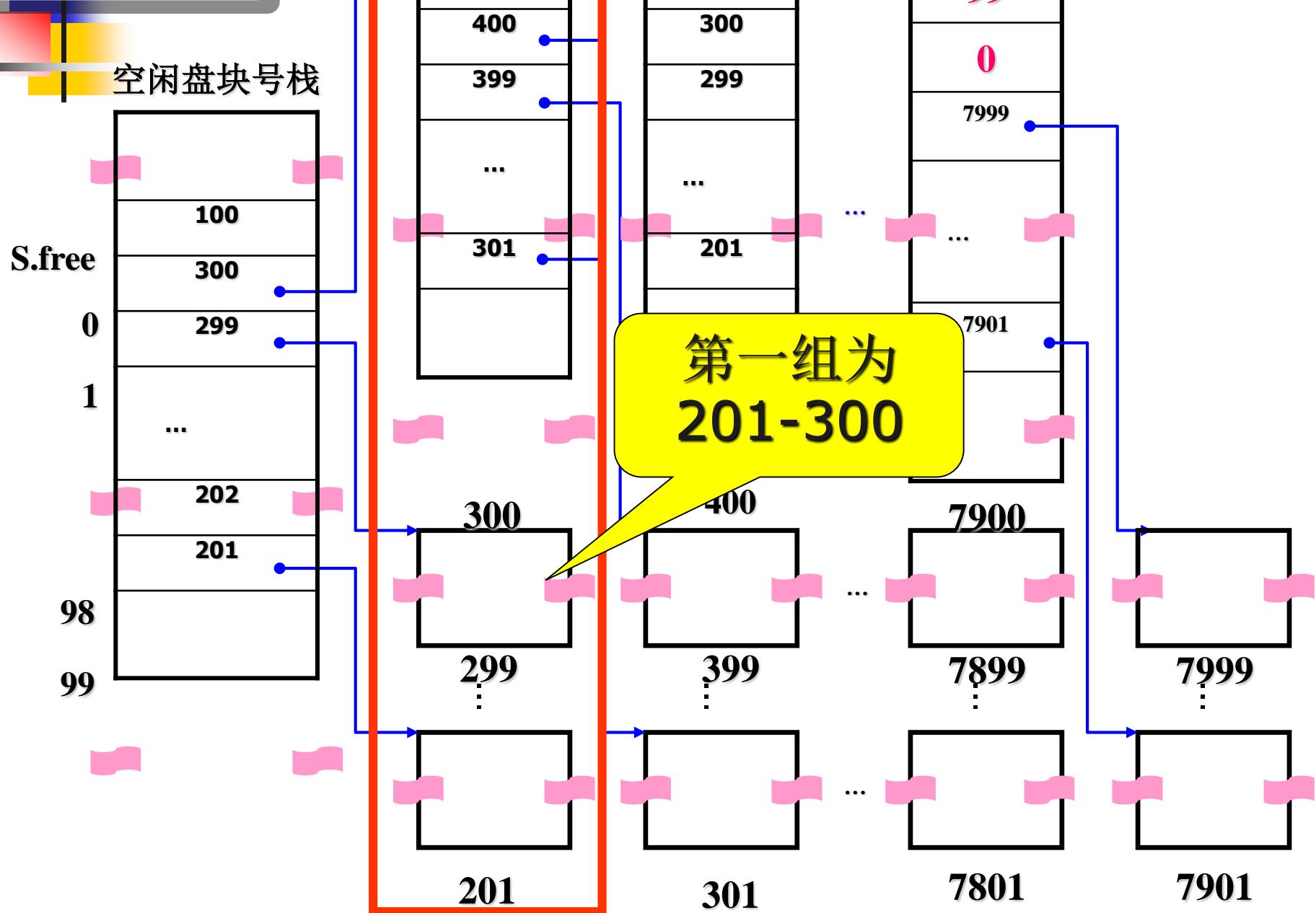
7901

空闲盘块的组织

文件存储空间中的所有空闲盘块，被分成若干个组。比如，将每100个盘块作为一组，假定盘上共有10000个盘块，将会分成100个组，其中第201-7999号盘块用于存放文件，即作为文件区

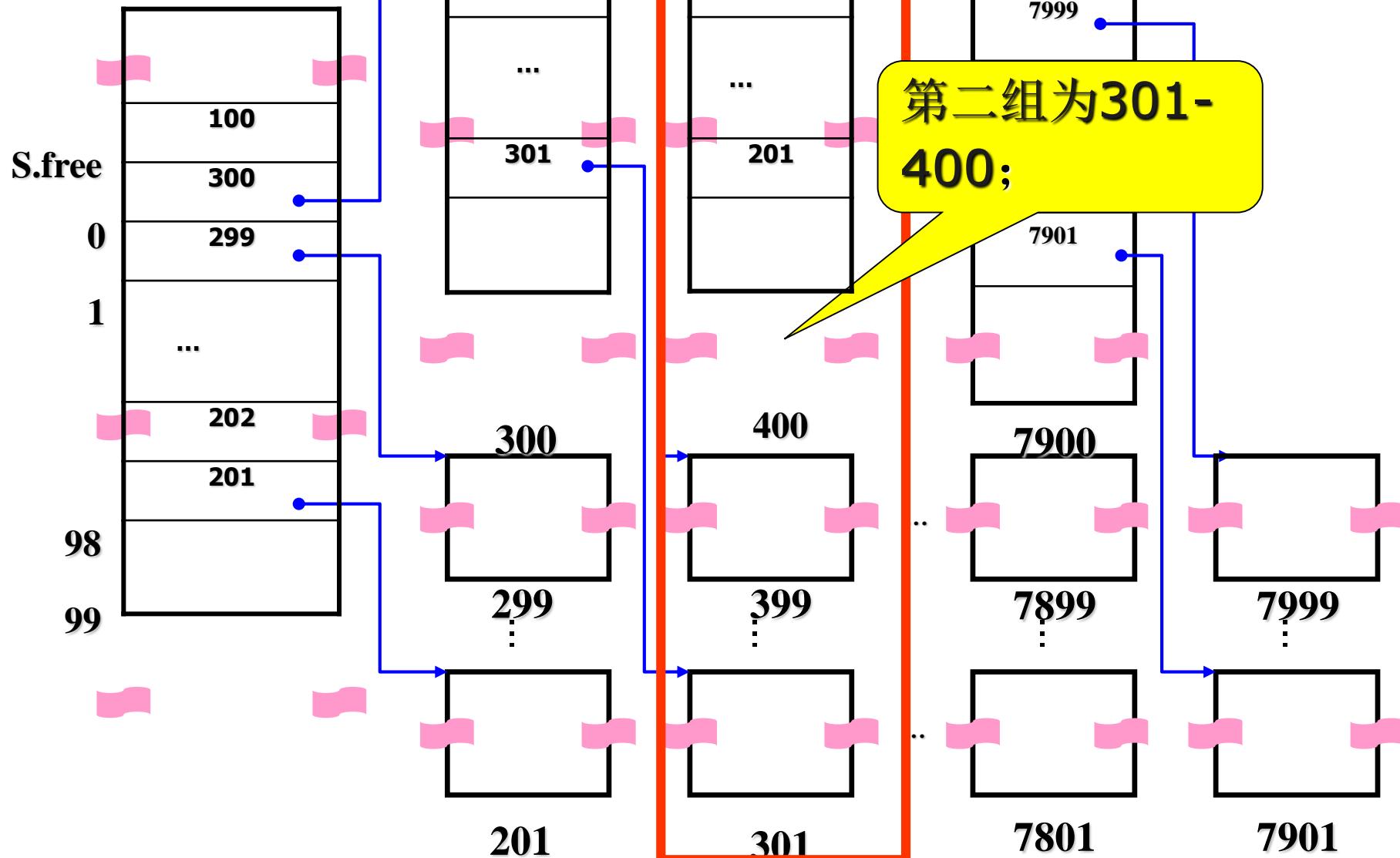


空闲盘块的组织



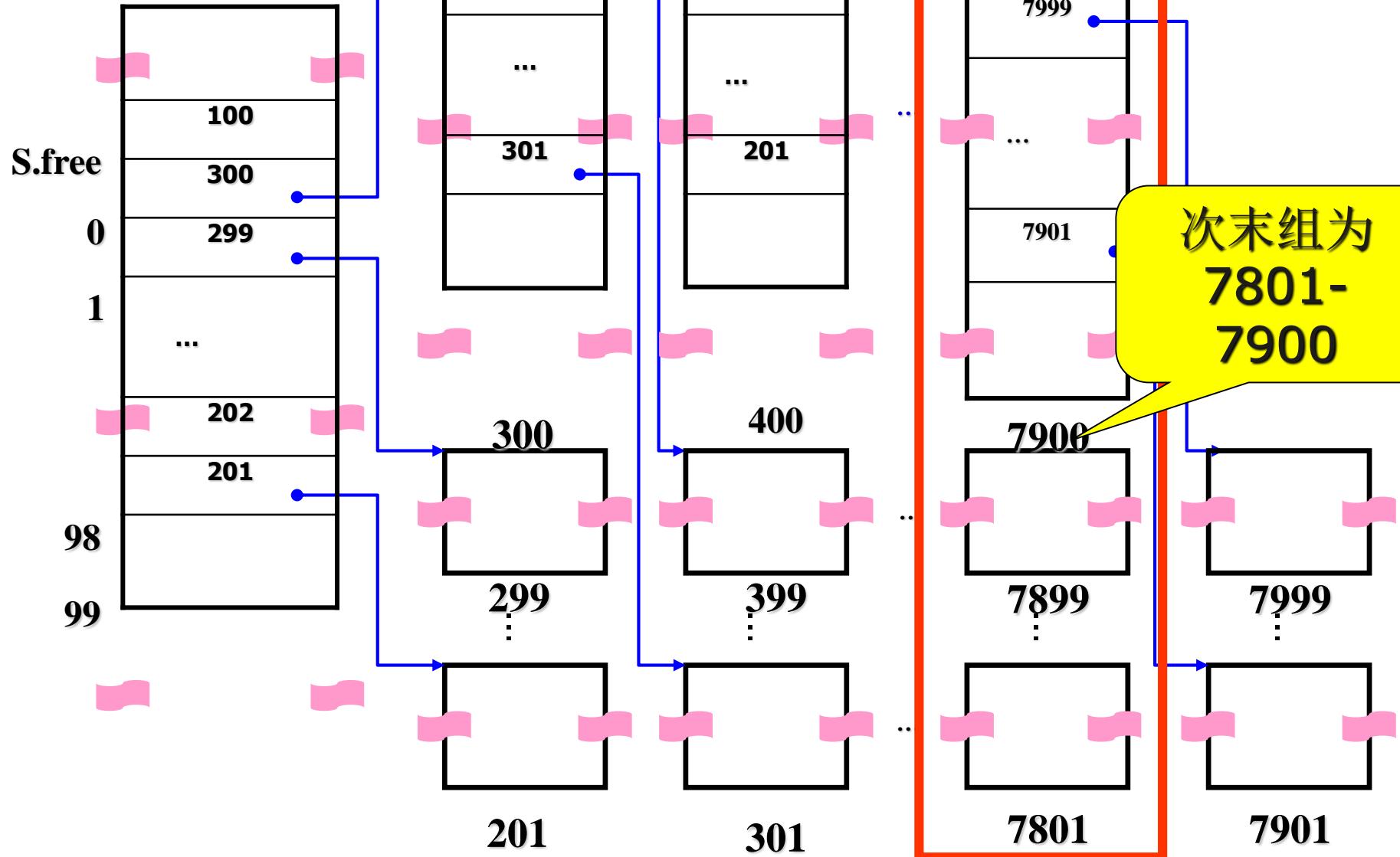
空闲盘块的组织

空闲盘块号栈



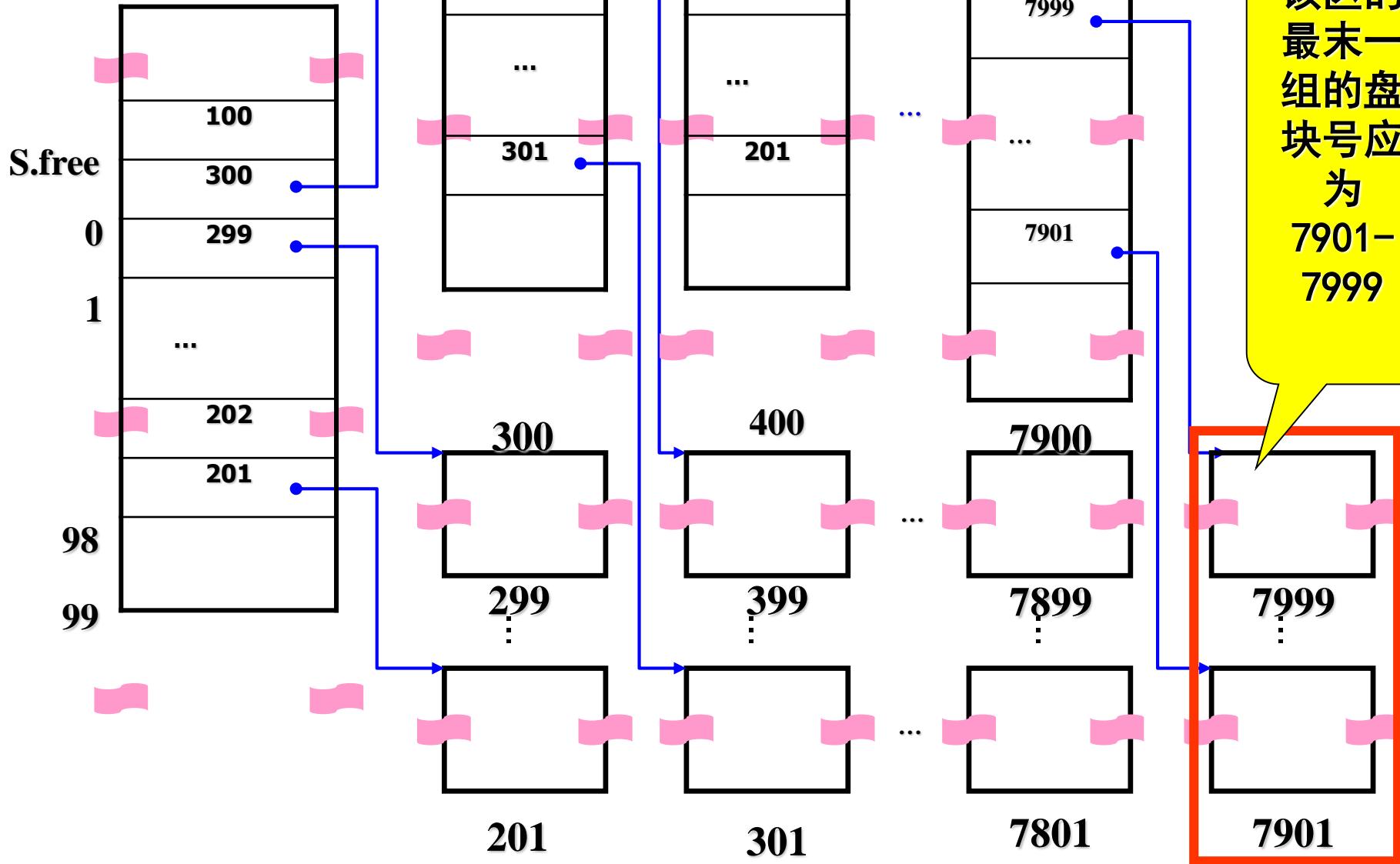
空闲盘块的组织

空闲盘块号栈

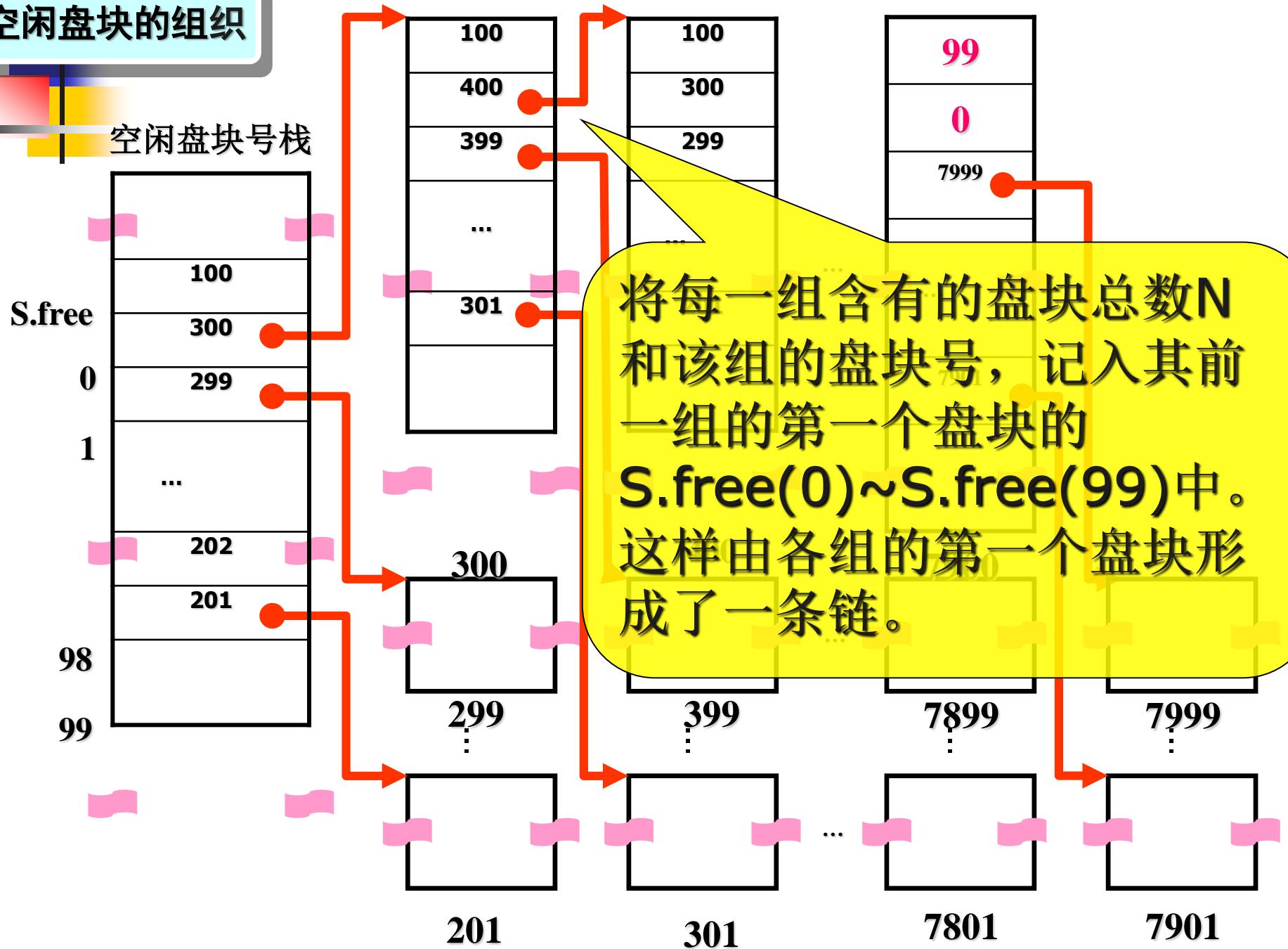


空闲盘块的组织

空闲盘块号栈

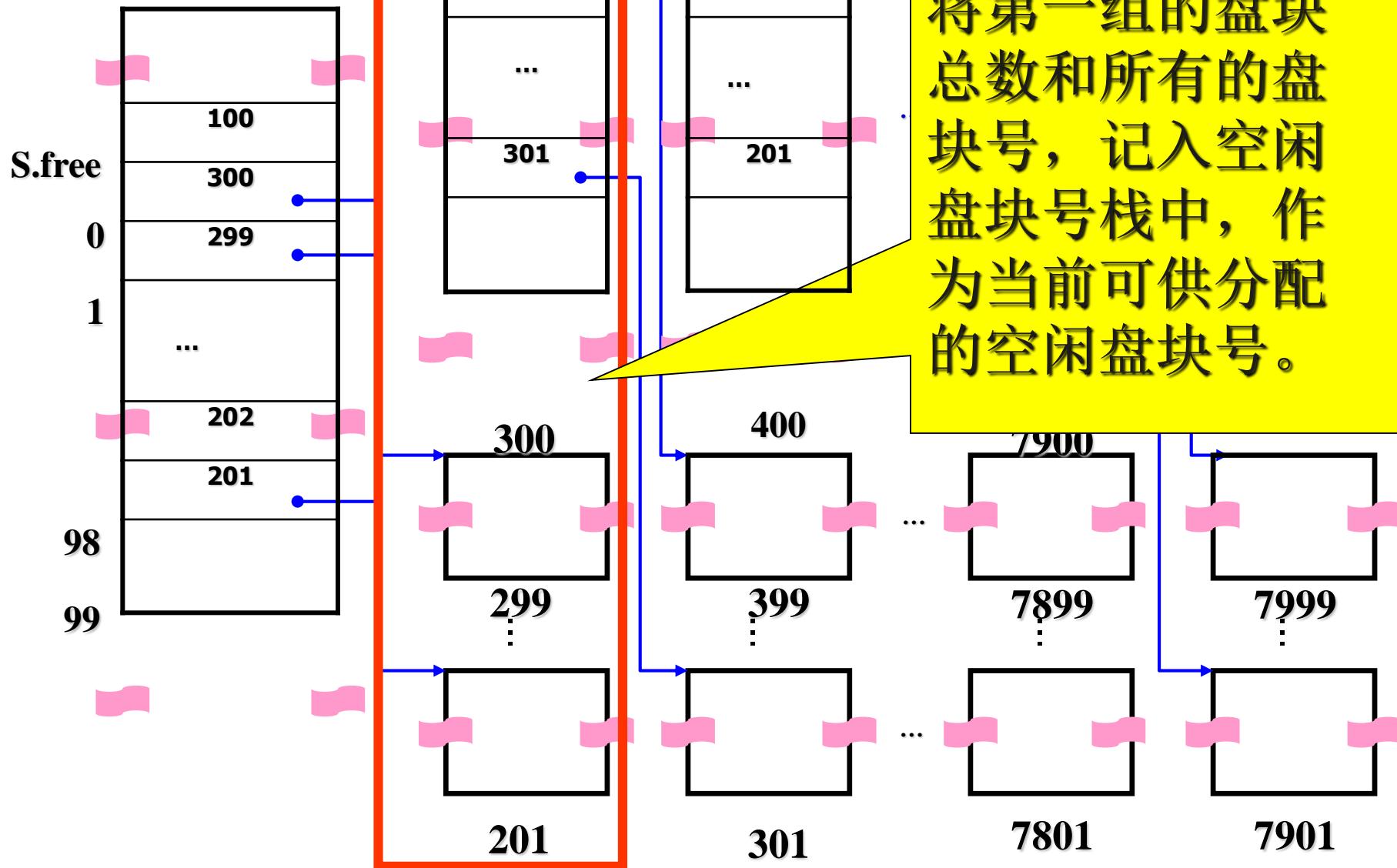


空闲盘块的组织



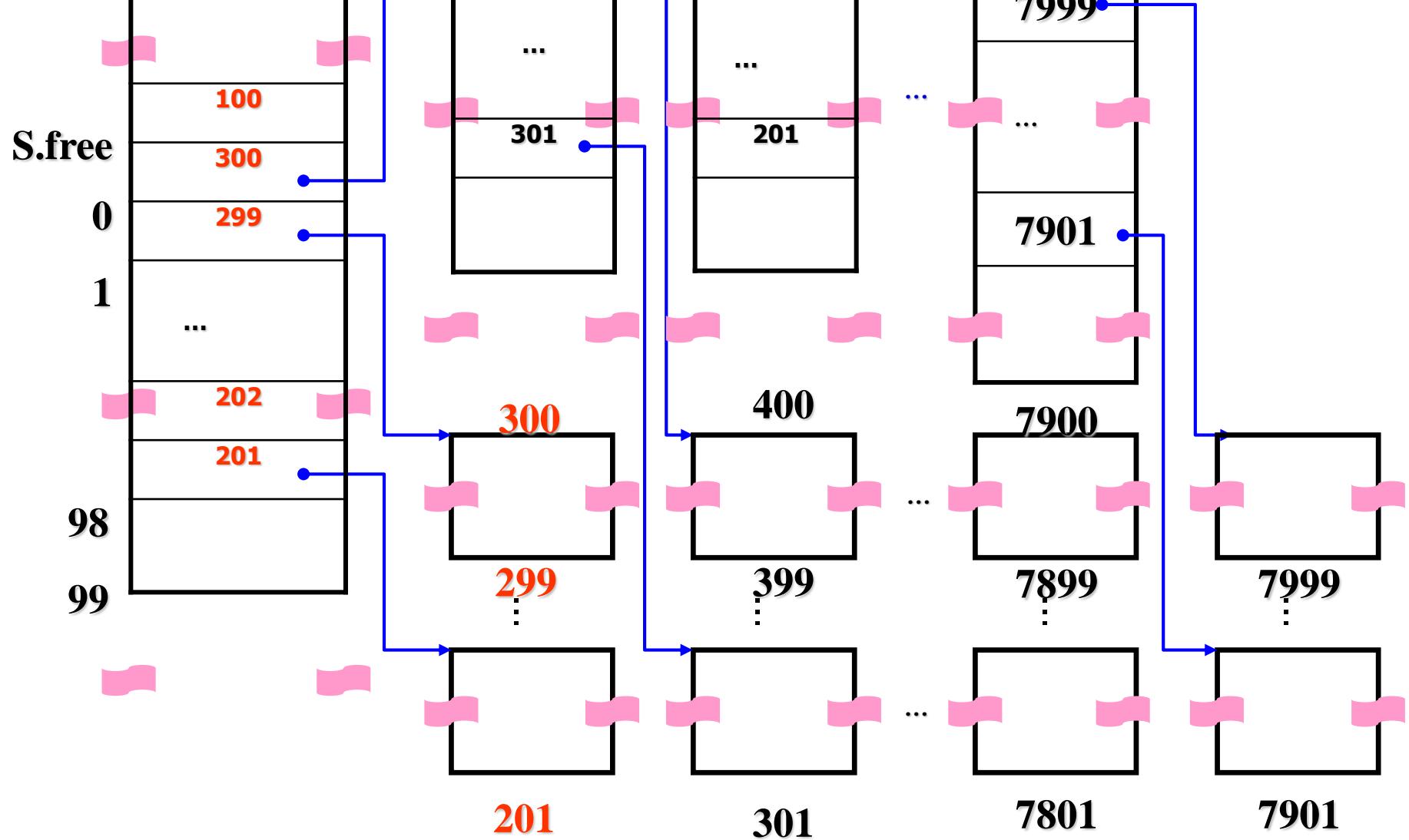
空闲盘块的组织

空闲盘块号栈



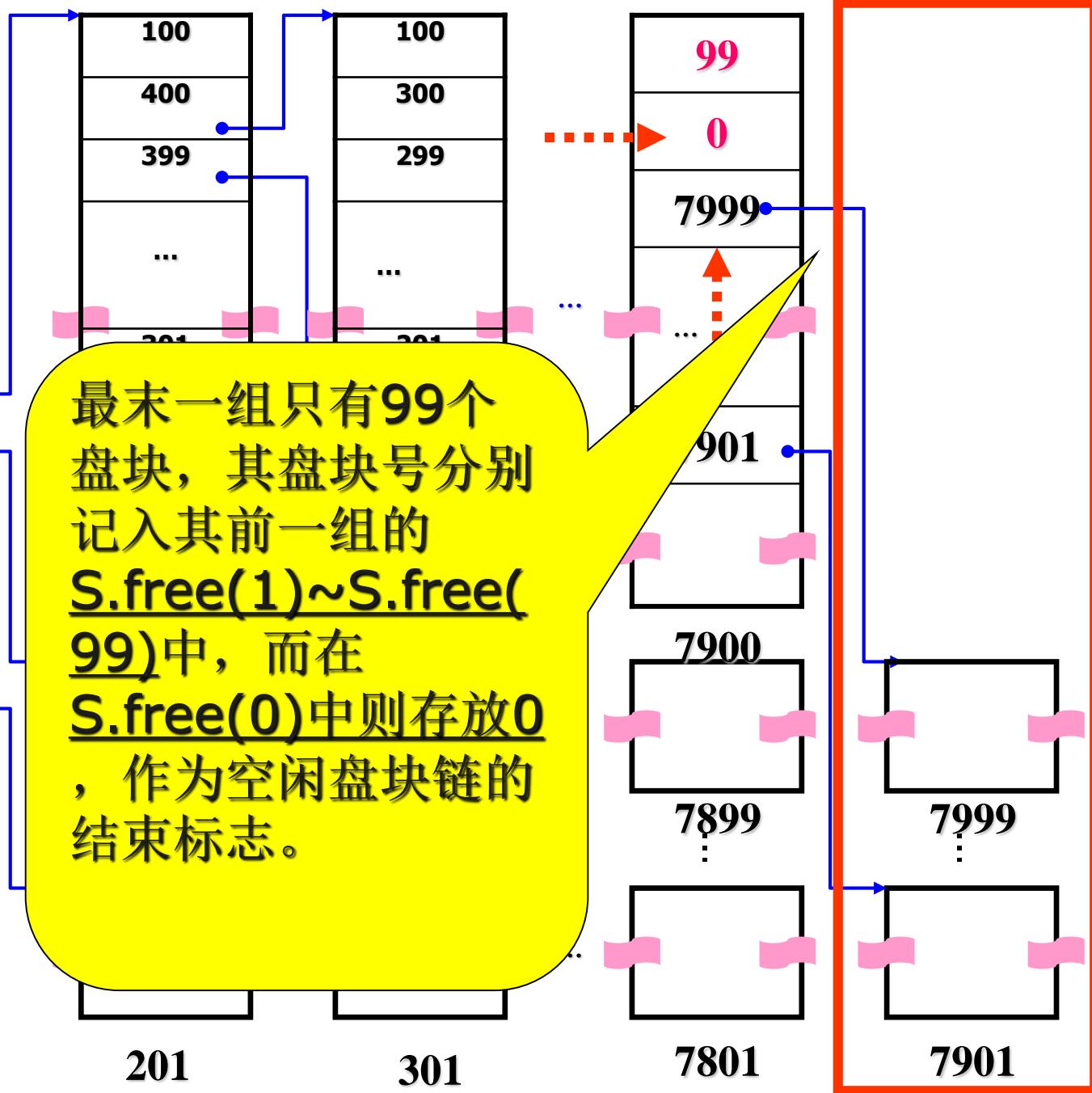
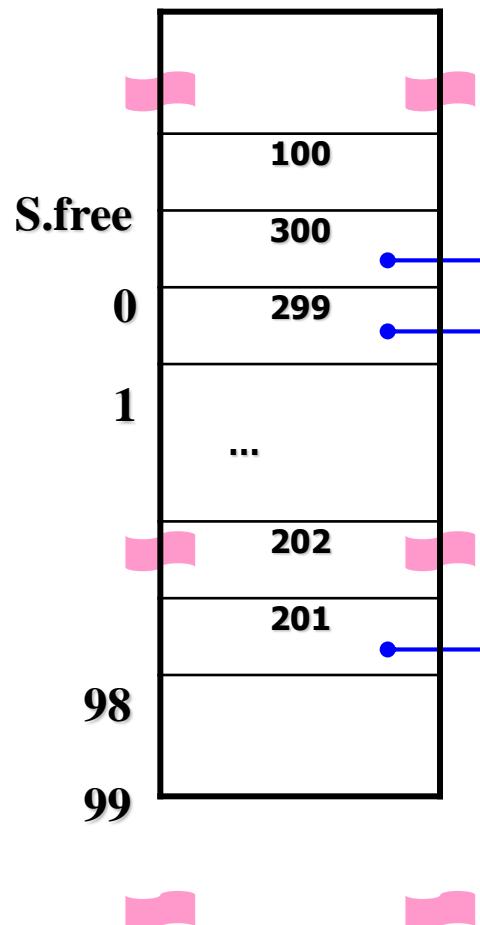
空闲盘块的组织

空闲盘块号栈

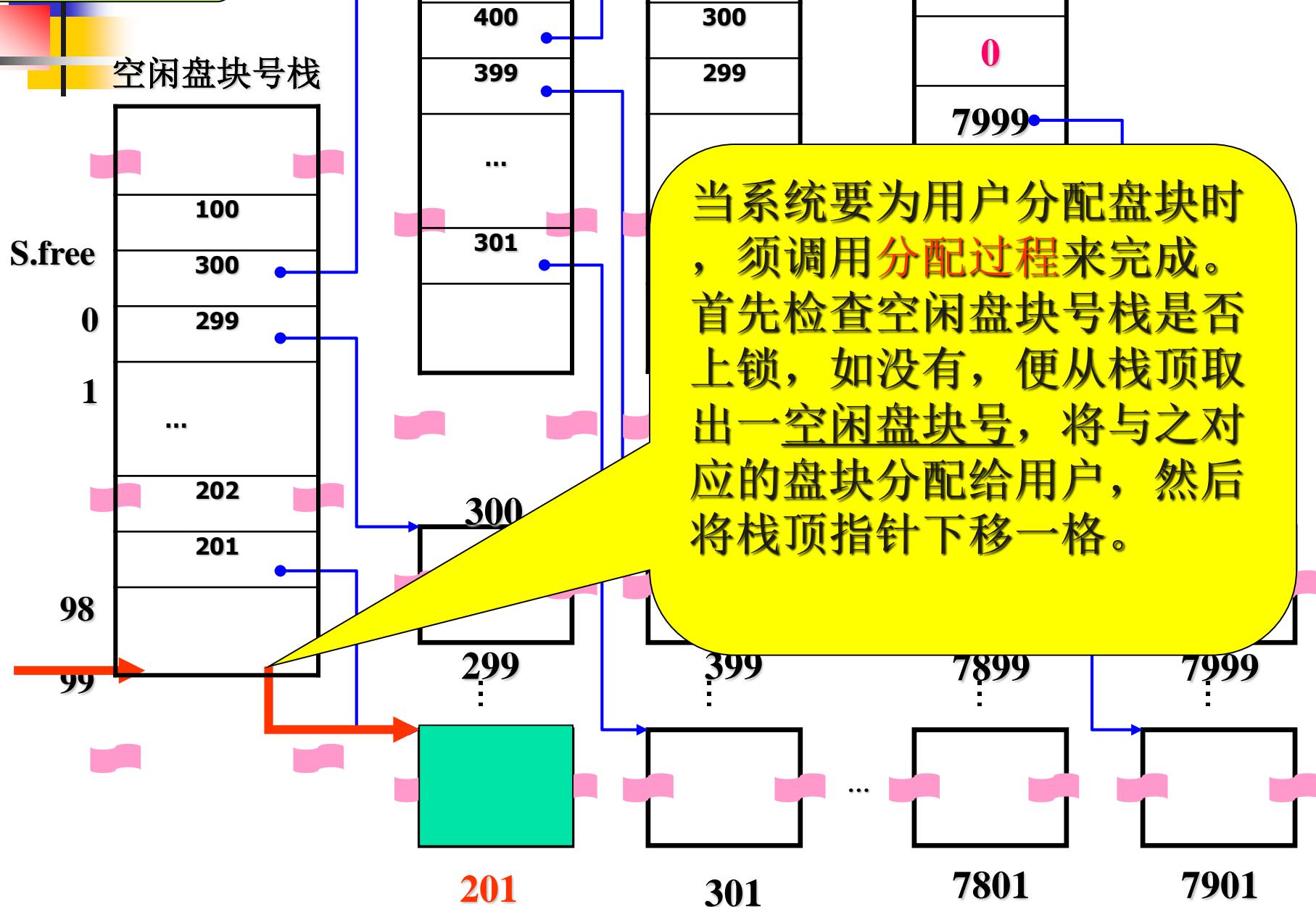


空闲盘块的组织

空闲盘块号栈

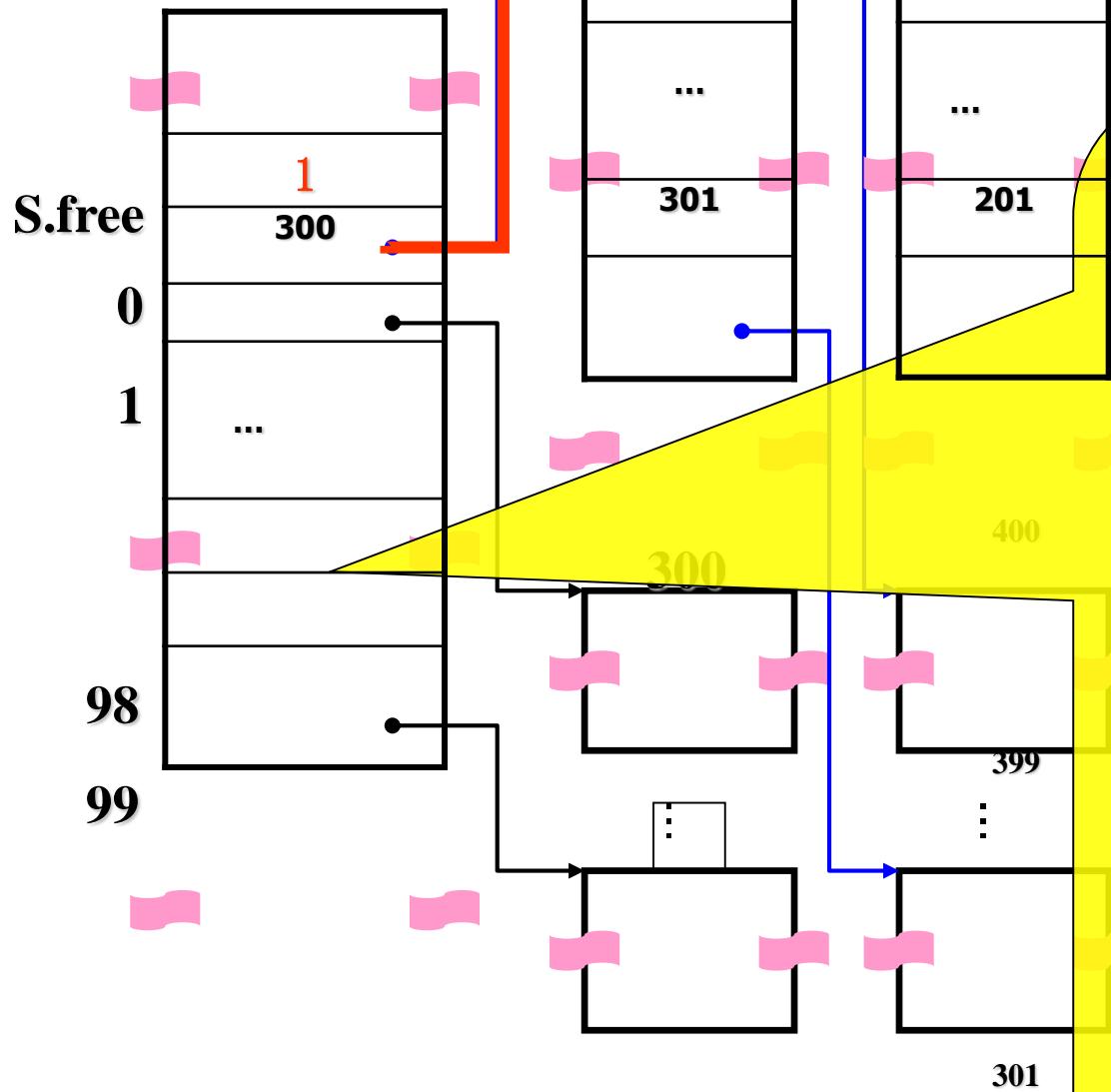


分配



分配

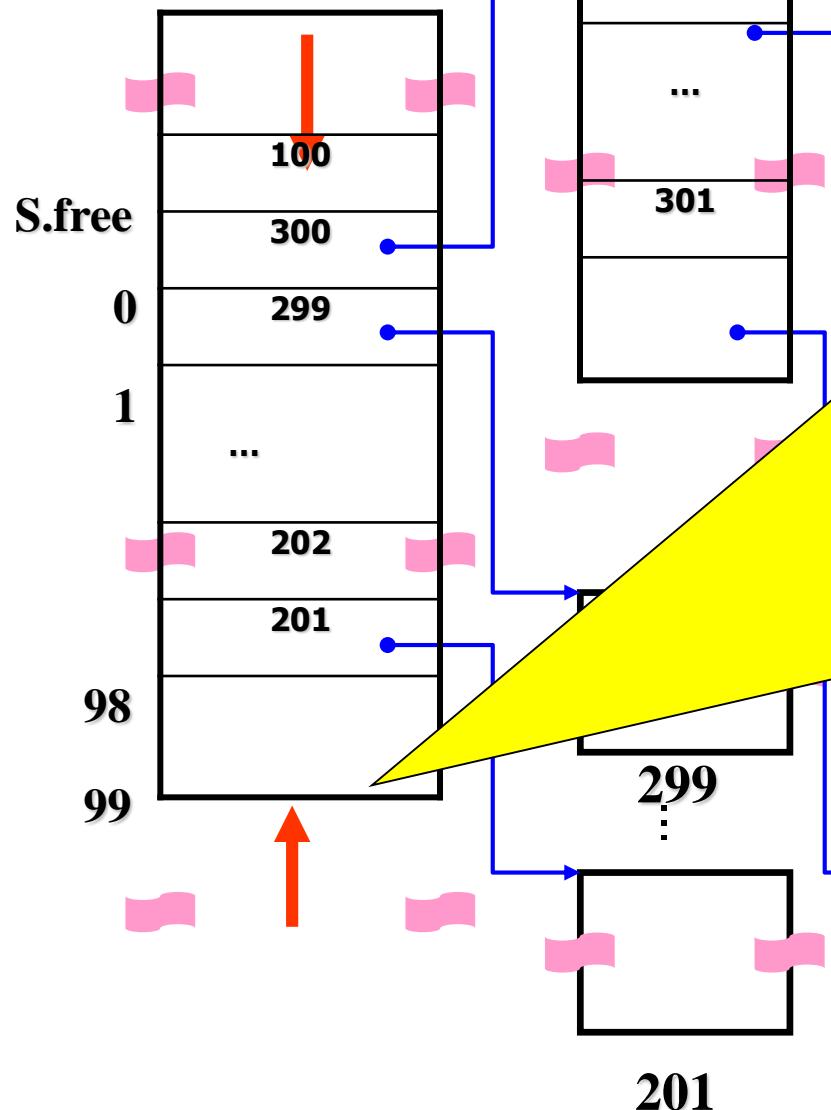
空闲盘块号栈



若该盘块号已是栈底，即 $S.\text{free}(0)$ ，这是当前栈中最后一个可供分配的盘块号。由于在该盘块号所对应的盘块中记有下一组可用的盘块号，因此须调用磁盘读过程，将该盘块内容读入栈中，作为盘块号栈的新内容，并把原栈底盘块分配出去。

回收

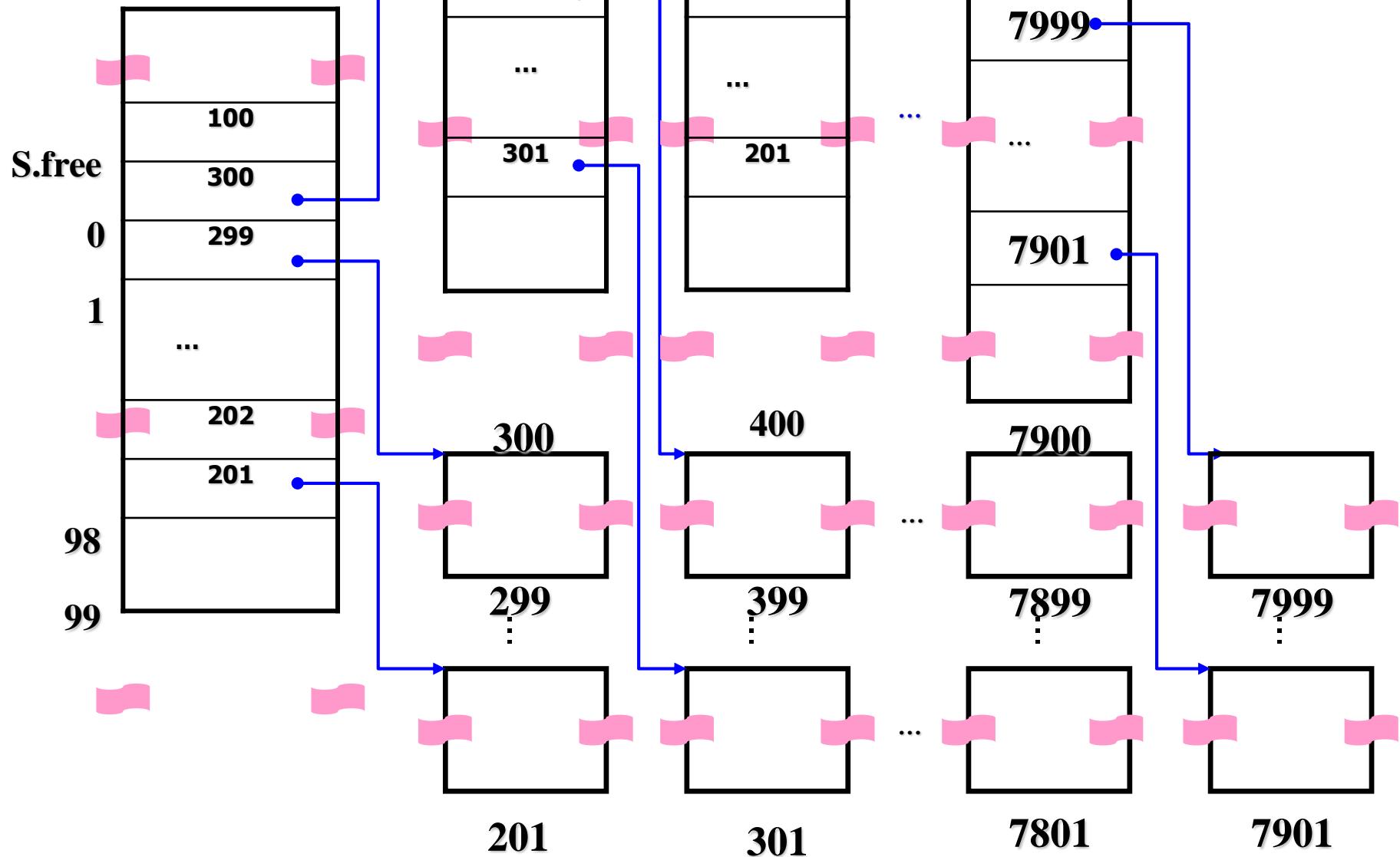
空闲盘块号栈



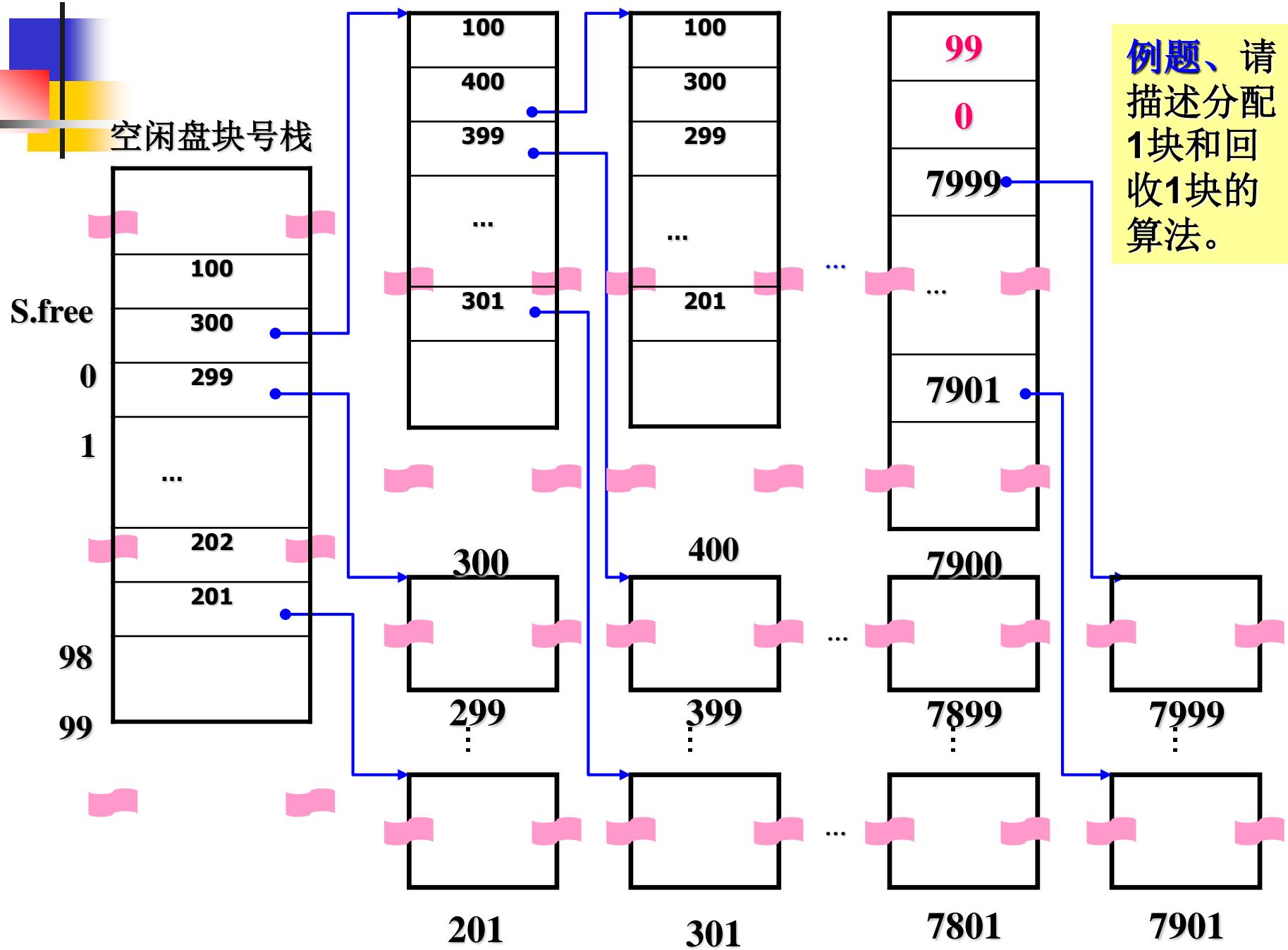
在系统回收空闲盘块时，须调用盘块回收过程进行回收。它是将回收盘块的盘块号记入空闲盘块号栈的顶部，并执行空闲盘块号加1操作。当栈中空闲盘块号数自己达100时，表示栈已满，便将现有栈中的100个盘块号，记入新回收的盘块中，再将其盘块号作为新栈底。

总结

空闲盘块号栈

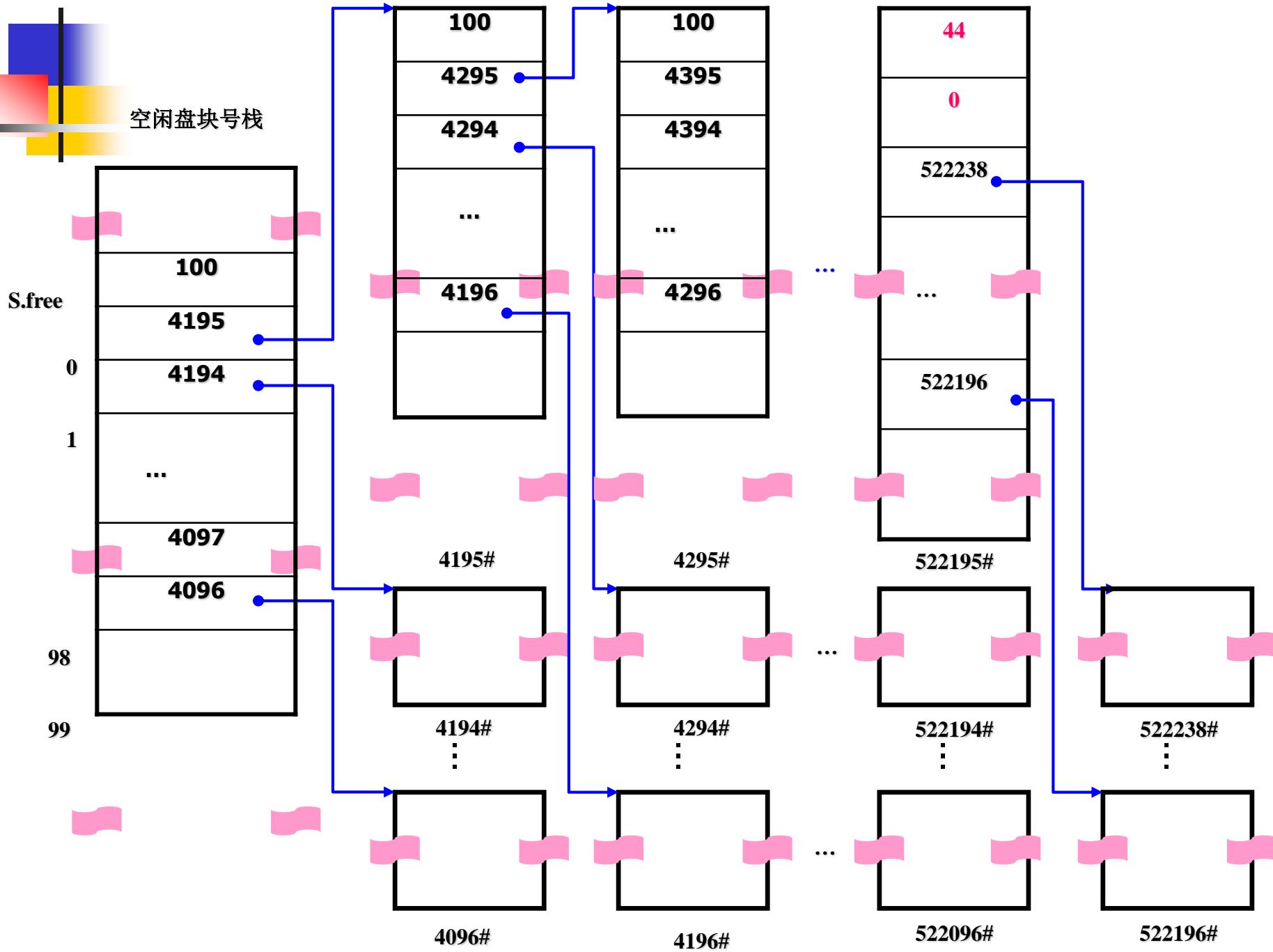


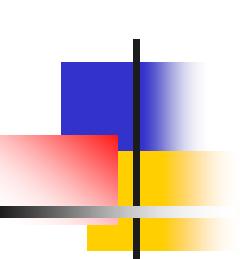
**例题、请
描述分配
1块和回
收1块的
算法。**



例题

- 计算机系统磁盘容量为520MB， 盘块大小为1KB。其中前4MB用于存放索引结点等，后10MB用作对换区，采用成组链接法管理外存空间，每组100个盘块。试画出外存尚未使用的成组链接图。
- 根据题意，该计算机系统尚未使用的外存空间为（ $520\text{MB} - 4\text{MB} - 10\text{MB} = 506\text{MB}$ ），即506K（也就是518144）个盘块，其盘块号为4K#～（510K-1）#（即4096#～522238#）。而每组100个盘块，故共有5182个盘块组，其中最后一个盘块组含43个盘块。因此，外存尚未使用的成组链接图如下所示：





第八章 磁盘存储器的管理

8.1 外存组织方式

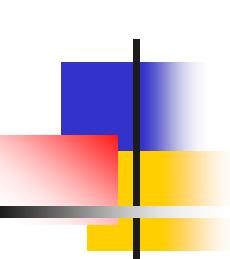
8.2 文件存储空间的管理

8.3 提高磁盘I/O速度的途径

8.4 提高磁盘可靠性的技术

8.3 提高磁盘I/O速度的途径

- 文件系统的性能可表现在多个方面，其中至关重要的一个方面是对文件的访问速度。为了提高对文件的访问速度，可以从三方面着手：
 - 改进文件的目录结构以及检索目录的方法来减少对目录的查找时间。
 - 选取好的文件存储结构，以提高对文件的访问速度。
 - 提高磁盘的I/O速度，能将文件中的数据快速地从磁盘传送
到内存中，或者相反。
- 提高磁盘I/O速度的途径
 - 磁盘高速缓存
 - 提高磁盘I/O速度其它方法
 - 廉价磁盘冗余阵列



第八章 磁盘存储器的管理

8.1 外存组织方式

8.2 文件存储空间的管理

8.3 提高磁盘I/O速度的途径

8.4 提高磁盘可靠性的技术

8.4 提高磁盘可靠性的技术

- 影响文件安全性的主要因素有人为因素、系统因素和自然因素三类，为了确保文件系统的安全性应采取三方面的措施：
 - 采取存取控制机制可以防止人为因素造成文件的不安全性；
 - 通过磁盘容错技术来防止由系统因素造成文件不安全性；
 - 采用建立“后备系统”的方法来防止由自然因素所造成的不安全性。
- 磁盘容错技术
 - 第一级容错技术：是最基本的一种磁盘容错技术，主要用于防止因磁盘表面缺陷所造成的数据丢失。它包含双份目录、双份文件分配表及写后校验等措施。
 - 第二级容错技术：主要用于防止由磁盘驱动器和磁盘控制器故障所导致的系统不能正常工作，它具体可分为磁盘镜像与磁盘双工。
 - 第三级容错技术：是系统容错技术，它基于集群技术实现容错。

基于集群技术的容错功能

- 所谓集群是指由一组互连的自主计算机组成统一的计算机系统，给人们的感觉是它们是一台机器。
- 利用集群系统不仅可提高系统的并行处理能力，还可用于提高系统的可用性。
- 集群系统的工作模式有三种：
 - 双机热备份模式：备有两台服务器，两者的处理能力通常完全相同的，一台作为主服务器，另一台作为备份服务器。
 - 双机互为备份模式：两台服务器均为在线服务器，它们各自完成自己的任务。在这种模式中，最好在每台服务器内都配置两台硬盘，一个用于装载系统程序和应用程序，另一个用于接收由另一台服务器发来的备份数据，作为该服务器的镜像盘。
 - 公共磁盘模式：为了减少信息复制的开销，可以将多台计算机连接到一台公共的磁盘系统上去。