|  |  |  |
| --- | --- | --- |
| RAID 0  (NO redundancy) | 1. 將資料分散到不同的硬碟中，強迫一次的資料存取會同時存取多個硬碟 2. 為非容錯硬碟列 3. 只要一個硬碟發生錯誤責資料都會遺失 |  |
| RAID1  (mirroring) | 1. 容錯硬碟最普遍的做法 2. 需要用到RAID 0兩倍的硬碟數目 3. 當有資料寫入，這些資料也會被寫入另一個redundant disk(資料備份)，當發生錯誤可到mirror中讀取資料 4. 最昂貴的做法 |  |
| RAID 2  (error detecting and correcting code) | 1. 使用hamming code 2. 對較小的word有非常高的ECC(error correcting code) 3. 寫入時要讀出所有的硬碟資料，算hamming code在寫入ECC硬碟，效率不佳 4. 硬體成本第二高 5. 還原速度快 |  |
| RAID 3  (bit interleaved parity) | 1. availability的成本可以降到1/N (N: protection group中的硬碟數目) 2. 不做資料備份，加入足夠的資訊來還原錯誤時遺失的資料 3. 對同一protection group的硬碟作讀寫時，會有額外硬碟儲存檢查資訊  * Parity(同位)—一種錯誤更正法，當錯誤發生從同位硬碟減去正常硬碟的資料，就是錯誤資料  1. 花費較多時間在恢復資料，花費較少的儲存裝置 2. 每一筆存取都會動用到所有硬碟 |  |
| Raid 4  (block interleaved parity) | 1. 不管是small-read /small-write都可平行處理 2. 和RAID 3有相同的資料硬碟和檢查硬碟比例，但存取方式不同 3. 同位(parity)以block方式被儲存 4. 只對保護群組中的一個硬碟做少量存取，大量存取必須存取所有位於同一保護群組的硬碟 5. 對於small-write必須存取所有剩餘硬碟來重新計算同位 6. 把同位當作一組資料總和，藉由觀察位元的改變，當寫入新資料時只需改變同位硬碟上相對應位元，使得在small-write時能減少讀取硬碟個數 7. 有效率的支援大量讀取、大量寫入和小量讀取的混合操作，允許小量寫入   缺點：同位必須在每次寫入時不斷被更新，導致同位硬碟成為連續寫入的瓶頸 |  |
| RAID 5  (distributed block interleaved parity) | 1. 分散式的同位組織 2. 為了修正RAID 4的瓶頸，把同位資訊分散到所有的硬碟中，消除單一的硬碟瓶頸 |  |
| RAID 6  (P+Q Redundancy) | 1. 同位的方式保護單一硬碟失效 2. 當單一的失效修正不夠時，第二次計算的同位可以包含全部資料和另一個檢查硬碟的內容，第二個檢驗區塊能修復第二個失效 3. 檢查硬碟是RAID 5的兩倍 |  |
| RAID 1+0 (striped mirror)  RAID 0+1 (mirrored stripe) |  | |

* RAID 1及RAID 5被廣泛的使用在伺服器上(80%採用)

|  |  |  |  |
| --- | --- | --- | --- |
| 匯流排仲裁  (決定哪個bus master下次可以使用bus的方式) | 菊花鏈仲裁  (daisy chain arbitration) | 1. 從優先權最高的排到優先權最低的，並依序執行 2. 想要存取匯流排的高優先權裝置，只需要攔截匯流排同意訊號，不讓低優先權裝置看到   優點：簡單  缺點：不公平   * 為了增進公平性， 規定剛使用過的不能重新要求匯流排，除非匯流排要求線降下 |  |
| 集中式、平行是仲裁法  (centralized、parallel arbitration) | 1. 使用多個要求線並且由裝置個別提出匯流排要求 2. 從這些要求匯流排的裝置中選取一個最匯流排的主控者   缺點：需要集中式仲裁器，可能造成匯流排使用的瓶頸   * PCI就是集中式仲裁法 |  |
| 自身選擇的分散式仲裁法  (distributed arbitration by self-selection) | 1. 採用多個要求線，參與要求匯流排的裝置可決定誰獲得匯流排使用權 2. 每個想要得到匯流排使用權的裝置會廣播其使用匯流排的優先順序 3. 在固定時間間隔後想要得匯流排使用權的裝置，藉由檢查手邊所收到的優先順序號碼便可決定 4. 每個裝置可自行判斷是否為最高優先權的要求  * Apple MAC II的NuBus使用此架構 |  |
| 衝突偵測分散式仲裁法  (distributed arbitration by collision detection) | 1. 每個裝置獨立要求匯流排 2. 多個同時要求會衝突 3. 可以偵測衝突並選擇這些衝突的使用者  * 乙太網路使用此法 |  |