其他附註

Enterprise applications

NetApp
May 09, 2024

目錄

其他附註

Oracle 資料庫效能最佳化與基準測試程序

準確測試資料庫儲存效能是極為複雜的主題。需要瞭解下列問題：

- IOPS 與處理量
- 前景與背景 I/O 作業之間的差異
- 延遲對資料庫的影響
- 許多作業系統和網路設定也會影響儲存效能

此外、還有非儲存資料庫工作要考量。最佳化儲存效能並不會帶來實用效益、因為儲存效能不再是效能的限制因素。

大多數資料庫客戶現在都選擇 All Flash Array 、這會造成一些額外考量。例如、請考慮在雙節點 AFF A900 系統上進行效能測試：

- 有了 80/20 讀取 / 寫入比率、兩個 A900 節點可在延遲甚至超過 150 μ s 標記之前、提供超過 1M 的隨機資料庫 IOPS 。這遠遠超出了大多數資料庫目前的效能需求、很難預測預期的改善。儲存設備將會大幅清除、成為瓶頸。
- 網路頻寬是效能限制的常見來源。例如、旋轉式磁碟解決方案通常是資料庫效能的瓶頸、因為 I/O 延遲非常高。當 All Flash 陣列移除延遲限制時、障礙會頻繁移轉至網路。在虛擬化環境和刀鋒系統中、這一點尤其顯著、因為這些環境和刀鋒系統的真正網路連線能力難以視覺化。如果由於頻寬限制而無法充分利用儲存系統本身、這可能會使效能測試變得複雜。
- 由於 All Flash 陣列的延遲大幅改善、因此一般無法將 All Flash 陣列與含有旋轉磁碟的陣列進行效能比較。測試結果通常沒有意義。
- 將尖峰 IOPS 效能與 All Flash 陣列進行比較通常並不實用、因為資料庫不受儲存 I/O 限制例如、假設一個陣列可維持 500K 的隨機 IOPS 、而另一個陣列則可維持 300K 。如果資料庫花費 99% 的時間處理 CPU 、這種差異在現實世界中是不相關的。工作負載永遠不會使用儲存陣列的完整功能。相反地、尖峰 IOPS 功能在整合平台中可能非常重要、而在整合平台中、儲存陣列預期會載入至尖峰容量。
- 請務必在任何儲存測試中考慮延遲和 IOPS 。市面上許多儲存陣列都宣稱 IOPS 極高、但延遲卻使這些 IOPS 在這類層級上無法使用。使用 All Flash Array 的典型目標是 1 毫秒標記。更好的測試方法不是測量最大可能的 IOPS 、而是判斷儲存陣列在平均延遲大於 1 毫秒之前可以維持多少 IOPS 。

Oracle 自動工作負載儲存庫與基準測試

Oracle 效能比較的黃金標準是 Oracle 自動工作負載儲存庫（ AWR ）報告。

有多種類型的 AWR 報告。從儲存點來看、執行所產生的報告 `awrrpt.sql` Command 是最全面且最有價值的命令、因為它針對特定資料庫執行個體、並包含一些詳細的分佈圖、可根據延遲來中斷儲存 I/O 事件。

比較兩個效能陣列的理想方法是在每個陣列上執行相同的工作負載、並產生精確鎖定工作負載的 AWR 報告。在執行時間極長的工作負載中、可以使用包含開始和停止時間的單一 AWR 報告、但最好將 AWR 資料分成多份報告。例如、如果批次工作從午夜執行至上午 6 點、請建立一系列從午夜– 1 點、上午 1 點– 2 點開始的一小時 AWR 報告。

在其他情況下、應最佳化非常簡短的查詢。最佳選項是以查詢開始時建立的 AWR 快照為基礎的 AWR 報告、以

及在查詢結束時建立的第二個 AWR 快照。否則資料庫伺服器應保持安靜、以將會使分析中查詢活動模糊的背景活動降至最低。

如果無法取得 AWR 報告、 Oracle 狀態報告是一個很好的替代方案。它們包含與 AWR 報告相同的大部分 I/O 統計資料。

Oracle AWR 與疑難排解

AWR 報告也是分析效能問題的最重要工具。

與基準測試一樣、效能疑難排解也需要您精確測量特定工作負載。如果可能、請在向 NetApp 支援中心回報效能問題、或與 NetApp 或合作夥伴客戶團隊合作、討論新解決方案時提供 AWR 資料。

提供 AWR 資料時、請考量下列需求：

- 執行 `awrrpt.sql` 產生報告的命令。輸出可以是文字或 HTML 。
- 如果使用 Oracle Real Application Clusters （ RAC ）、請為叢集中的每個執行個體產生 AWR 報告。
- 鎖定問題存在的特定時間。AWR 報告的最長可接受使用時間通常為一小時。如果問題持續數小時、或涉及多小時作業、例如批次工作、請提供多個一小時的 AWR 報告、涵蓋整個分析期間。
- 如有可能、請將 AWR 快照時間間隔調整為 15 分鐘。此設定可執行更詳細的分析。這也需要額外執行 `awrrpt.sql` 提供每 15 分鐘間隔的報告。
- 如果問題是非常短的執行查詢、請根據作業開始時建立的 AWR 快照、以及作業結束時建立的第二個 AWR 快照、提供 AWR 報告。否則、資料庫伺服器應保持安靜、以將會使分析中作業的活動受到影響的背景活動減至最低。
- 如果在特定時間回報效能問題、但未在其他時間回報、請提供額外的 AWR 資料、以展現良好的效能來進行比較。

calibr_IO

。 `calibrate_io` 絕對不可使用命令來測試、比較或基準測試儲存系統。如 Oracle 文件所述、本程序會校正儲存設備的 I/O 功能。

校準與基準測試不同。此命令的目的是發佈 I/O 、藉由最佳化發行給主機的 I/O 層級、協助校正資料庫作業並改善其效率。因為執行的 I/O 類型 `calibrate_io` 作業並不代表實際的資料庫使用者 I/O 、結果無法預測、而且經常無法重現。

SLOB2

SLOB2 是愚蠢的小 Oracle 基準測試工具、已成為評估資料庫效能的首選工具。這是由 Kevin Closson 開發的、可在取得 "https://kevinclosson.net/slob/"。安裝和設定需要幾分鐘的時間、它使用實際的 Oracle 資料庫來在使用者可定義的資料表空間上產生 I/O 模式。這是少數幾種可用的測試選項之一、可將全快閃陣列與 I/O 飽和它也有助於產生更低層級的 I/O 、以模擬低 IOPS 但對延遲敏感的儲存工作負載。

交換台工作台

交換基準台可用於測試資料庫效能、但使用交換基準台的方式會對儲存造成壓力、這是非常困難的。NetApp 尚未從 SwingWorkbench 中看到任何測試結果、這些測試產生足夠的 I/O 、使其在任何 AFF 陣列上都成為重大負載。在有限的情況下、訂單輸入測試（ OET ）可用於從延遲點評估儲存設備。這在資料庫對於特定查詢具有已知延遲相依性的情況下很有用。必須注意確保主機和網路已正確設定、以實現 All Flash 陣列的延遲潛力。

HammerDB

HammerDB 是一種資料庫測試工具、可模擬 TPC-C 和 TPC-H 基準測試等。建構足夠大的資料集以正確執行測試可能需要很長時間、但它可以是評估 OLTP 和資料倉儲應用程式效能的有效工具。

Orion

Oracle Orion 工具通常與 Oracle 9 搭配使用、但並未加以維護、以確保與各種主機作業系統的變更相容。由於與作業系統和儲存組態不相容、因此很少與 Oracle 10 或 Oracle 11 搭配使用。

Oracle 重新編寫了該工具，默認情況下，該工具與 Oracle 12c 一起安裝。雖然本產品已經過改良、並使用許多與實際 Oracle 資料庫相同的呼叫、但它並未使用 Oracle 所使用的相同程式碼路徑或 I/O 行為。例如、大部分的 Oracle I/O 都是同步執行、這表示資料庫會暫停、直到 I/O 作業在前景完成為止。只是以隨機 I/O 淹沒儲存系統、並不是真正的 Oracle I/O 複製、也無法提供直接的儲存陣列比較方法、也無法測量組態變更的影響。

也就是說、 Orion 有一些使用案例、例如一般測量特定主機網路儲存組態的最大可能效能、或是測量儲存系統的健全狀況。仔細測試後、只要參數包括 IOPS 、處理量和延遲的考量、並嘗試忠實複寫真實的工作負載、就能設計出可用的 Orion 測試來比較儲存陣列或評估組態變更的影響。

過時的 NFSv3 鎖定和 Oracle 資料庫

如果 Oracle 資料庫伺服器當機、則重新啟動時可能會發生過時的 NFS 鎖定問題。請仔細注意伺服器上的名稱解析設定、以避免此問題。

產生此問題的原因是建立鎖定和清除鎖定會使用兩種稍微不同的名稱解析方法。涉及兩個過程： Network Lock Manager （ NLM ）和 NFS 用戶端。NLM 使用 `uname n` 以決定主機名稱、而 `rpc.statd` 程序用途 `gethostbyname()`。這些主機名稱必須相符、作業系統才能正確清除過時的鎖定。例如、主機可能正在尋找擁有的鎖定 `dbserver5`**但鎖定已由主機登錄為** `dbserver5.mydomain.org`。如果 `gethostbyname()` 不會傳回與相同的值 `uname -a`，則鎖定釋放程序未成功。

下列範例指令碼會驗證名稱解析是否完全一致：

```
#! /usr/bin/perl
$uname=`uname -n`;
chomp($uname);
($name, $aliases, $addrtype, $length, @addrs) = gethostbyname $uname;
print "uname -n yields: $uname\n";
print "gethostbyname yields: $name\n";
```

如果 `gethostbyname` 不符 `uname`、可能是過時的鎖定。例如、此結果顯示潛在問題：

```
uname -n yields: dbserver5
gethostbyname yields: dbserver5.mydomain.org
```

解決方案通常是透過變更主機出現在中的順序來找到 `/etc/hosts`。例如、假設 hosts 檔案包含下列項目：

```
10.156.110.201  dbserver5.mydomain.org dbserver5 loghost
```

若要解決此問題、請變更完整網域名稱和簡短主機名稱出現的順序：

```
10.156.110.201  dbserver5 dbserver5.mydomain.org loghost
```

`gethostbyname()` 現在傳回短 `dbserver5` 主機名稱、符合的輸出 `uname`。因此、鎖定會在伺服器當機後自動清除。

Oracle 資料庫的 WAFL 對齊驗證

正確的 WAFL 對齊對於良好的效能至關重要。雖然 ONTAP 以 4KB 單位管理區塊、但這並不表示 ONTAP 以 4KB 單位執行所有作業。事實上、 ONTAP 支援不同大小的區塊作業、但基礎會計是由 WAFL 以 4KB 為單位進行管理。

「對齊」一詞是指 Oracle I/O 與這些 4KB 單元的相對應方式。最佳效能要求 Oracle 8KB 區塊位於磁碟機上的兩個 4KB WAFL 實體區塊上。如果區塊偏移 2KB 、則此區塊會位於一半的 4KB 區塊、一個獨立的完整 4KB 區塊、然後是第三個 4KB 區塊的一半。這種安排會導致效能降低。

對齊並不涉及 NAS 檔案系統。Oracle 資料檔案會根據 Oracle 區塊的大小、與檔案的開頭對齊。因此、 8KB 、 16KB 和 32KB 的區塊大小一律會對齊。所有區塊作業都會從檔案開頭偏移、單位為 4 KB 。

相反地、 LUN 在啟動時通常會包含某種驅動程式標頭或檔案系統中繼資料、以建立偏移。對齊在現代作業系統中很少是個問題、因為這些作業系統是專為可能使用原生 4KB 磁碟區的實體磁碟機所設計、因此也需要將 I/O 與 4KB 邊界對齊才能獲得最佳效能。

不過、有一些例外情況。資料庫可能已從未針對 4KB I/O 最佳化的舊版作業系統移轉、或是在建立分割區時發生使用者錯誤、可能導致偏移量、而大小單位不是 4KB 。

下列範例僅適用於 Linux 、但程序可適用於任何作業系統。

一致

以下範例顯示單一磁碟分割的單一 LUN 對齊檢查。

首先、建立使用磁碟機上所有可用分割區的分割區。

```
[root@host0 iscsi]# fdisk /dev/sdb
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF
disklabel
Building a new DOS disklabel with disk identifier 0xb97f94c1.
Changes will remain in memory only, until you decide to write them.
After that, of course, the previous content won't be recoverable.
The device presents a logical sector size that is smaller than
the physical sector size. Aligning to a physical sector (or optimal
I/O) size boundary is recommended, or performance may be impacted.
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-10240, default 1):
Using default value 1
Last cylinder, +cylinders or +size{K,M,G} (1-10240, default 10240):
Using default value 10240
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
[root@host0 iscsi]#
```

您可以使用下列命令以數學方式檢查對齊方式：

```
[root@host0 iscsi]# fdisk -u -l /dev/sdb
Disk /dev/sdb: 10.7 GB, 10737418240 bytes
64 heads, 32 sectors/track, 10240 cylinders, total 20971520 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 65536 bytes
Disk identifier: 0xb97f94c1
   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1              32    20971519    10485744   83  Linux
```

輸出顯示單位為 512 位元組、且分割區的開頭為 32 個單位。總共 32 x 512 = 16 、 834 位元組、這是 4KB WAFL 區塊的整數倍數。此分割區已正確對齊。

若要驗證正確的對齊方式、請完成下列步驟：

1. 識別 LUN 的通用唯一識別碼（ UUID ）。

```
FAS8040SAP::> lun show -v /vol/jfs_luns/lun0
              Vserver Name: jfs
                  LUN UUID: ed95d953-1560-4f74-9006-85b352f58fcd
                    Mapped: mapped`                 `
```

2. 進入 ONTAP 控制器上的節點 Shell 。

```
FAS8040SAP::> node run -node FAS8040SAP-02
Type 'exit' or 'Ctrl-D' to return to the CLI
FAS8040SAP-02> set advanced
set not found.  Type '?' for a list of commands
FAS8040SAP-02> priv set advanced
Warning: These advanced commands are potentially dangerous; use
         them only when directed to do so by NetApp
         personnel.
```

3. 在第一步中識別的目標 UUID 上開始收集統計資料。

```
FAS8040SAP-02*> stats start lun:ed95d953-1560-4f74-9006-85b352f58fcd
Stats identifier name is 'Ind0xffffff08b9536188'
FAS8040SAP-02*>
```

4. 執行一些 I/O請務必使用 `iflag` 用於確保 I/O 同步且無緩衝的引數。

   請務必小心使用此命令。反轉 `if` 和 `of` 引數會破壞資料。

```
[root@host0 iscsi]# dd if=/dev/sdb1 of=/dev/null iflag=dsync count=1000
bs=4096
1000+0 records in
1000+0 records out
4096000 bytes (4.1 MB) copied, 0.0186706 s, 219 MB/s
```

5. 停止統計資料並檢視對齊分佈圖。所有 I/O 都應位於 `.0` 貯體、表示 I/O 與 4KB 區塊邊界對齊。

```
FAS8040SAP-02*> stats stop
StatisticsID: Ind0xffffff08b9536188
lun:ed95d953-1560-4f74-9006-85b352f58fcd:instance_uuid:ed95d953-1560-
4f74-9006-85b352f58fcd
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.0:186%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.1:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.2:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.3:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.4:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.5:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.6:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.7:0%
```

未對齊

以下範例顯示 I/O 未對齊：

1. 建立不符合 4KB 邊界的分割區。這不是現代作業系統的預設行為。

```
[root@host0 iscsi]# fdisk -u /dev/sdb
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First sector (32-20971519, default 32): 33
Last sector, +sectors or +size{K,M,G} (33-20971519, default 20971519):
Using default value 20971519
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

2. 已建立磁碟分割、並使用 33 磁區偏移值、而非預設的 32 。重複中所述的程序 "一致"。直方圖顯示如下：

```
FAS8040SAP-02*> stats stop
StatisticsID: Ind0xffffff0468242e78
lun:ed95d953-1560-4f74-9006-85b352f58fcd:instance_uuid:ed95d953-1560-
4f74-9006-85b352f58fcd
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.0:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.1:136%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.2:4%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.3:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.4:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.5:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.6:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_align_histo.7:0%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:read_partial_blocks:31%
```

錯誤的對齊是顯而易見的。I/O 大多落在 * 之中 *.1 符合預期偏移的貯體。建立分割區時、它會比最佳化的預設值更進一步移入 512 個位元組、這表示長條圖偏移 512 個位元組。

此外 `read_partial_blocks` 統計資料為非零、這表示執行的 I/O 並未填滿整個 4KB 區塊。

重作記錄

此處說明的程序適用於資料檔案。Oracle 重做記錄和歸檔記錄檔有不同的 I/O 模式。例如、重做記錄是單一檔案的循環覆寫。如果使用預設的 512 位元組區塊大小、寫入統計資料看起來會像這樣：

```
FAS8040SAP-02*> stats stop
StatisticsID: Ind0xffffff0468242e78
lun:ed95d953-1560-4f74-9006-85b352f58fcd:instance_uuid:ed95d953-1560-4f74-
9006-85b352f58fcd
lun:ed95d953-1560-4f74-9006-85b352f58fcd:write_align_histo.0:12%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:write_align_histo.1:8%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:write_align_histo.2:4%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:write_align_histo.3:10%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:write_align_histo.4:13%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:write_align_histo.5:6%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:write_align_histo.6:8%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:write_align_histo.7:10%
lun:ed95d953-1560-4f74-9006-85b352f58fcd:write_partial_blocks:85%
```

I/O 會分散到所有分佈式分佈區、但這並不是效能考量。不過、重做記錄率極高可能會因為使用 4KB 區塊大小而受惠。在這種情況下、最好確定重做記錄 LUN 已正確對齊。不過、這對於資料檔案對齊的良好效能並不重要。