



## 如何發現系統 Resources 是否被鎖住

作者：倍力資訊第一事業部 系統顧問

張安翔 先生

### 前言

DEADLOCK 的發生對系統的性能有重要影響，經檢測發現，管理資訊系統的 DEADLOCK 主要是因為兩個或多個 User 搶佔同一表資料資源，引起長時間佔用同一資源，不是因為我們需要處理的交易太複雜，時間太長，而往往是因為我們在前端應用程式對資料庫作操作時忘了提交。本文將介紹如何發現系統資源被鎖定以及處理解決這種 DEADLOCK 的方法。

### 一、Sybase Lock 原理

資料共用與保持資料一致性本來就是相互矛盾的，而爲了要能夠享資料且又能維持資料的一致性，因而必須進行相對應的一些控制機制。這些控制機制的任務就是爲了避免共用衝突而引起的資料不一致。SYBASE ASE 控制機制就是加鎖機制（LOCKING）。

#### ► 鎖的類型

SYBASE ASE 有三種封鎖類型：排它鎖(exclusive lock, 簡稱 X 鎖)；共用鎖(share lock, 簡稱 S 鎖)；更新鎖 (update lock, 簡稱 U 鎖)，如下圖所示：

	Statements That Impose This Lock	If Data Already Has an S Lock	If Data Already Has an X Lock	If Data Already Has a U Lock	Granularities for Which This Lock Exists
Shared (S)	select	Place another S lock	Wait for lock to be released	Place another S lock	Row, page, table
Exclusive (X)	insert, update*, delete*	Wait for lock to be released	Wait for lock to be released	Wait for lock to be released	Row, page, table
Update (U)	update, delete	Place the U lock	Wait for lock to be released	Wait for lock to be released	Row, page

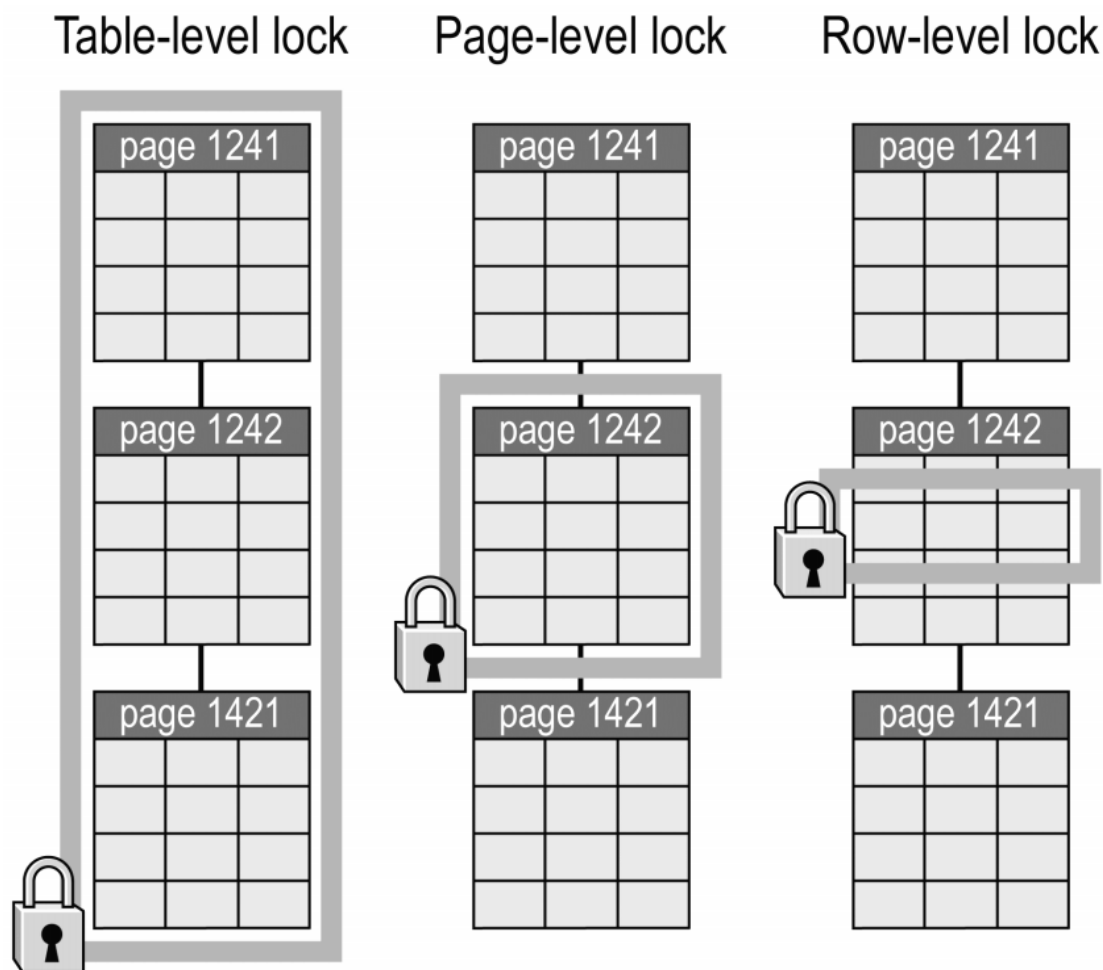
圖一：鎖的三種類型（資料來源：Sybase SGV2）

SYBASE ASE 是自動決定加鎖類型的。一般來說，讀（SELECT）操作使用 S 鎖，寫（UPDATE, INSERT 和 delete）操作使用 X 鎖。U 鎖是建立在頁級上的，它在一

個更新操作開始時獲得，當要修改這些頁時，U 鎖會升級為 X 鎖。

► 鎖的粒度 (Granularity of Locks)

Sybase ASE 有三級鎖：表鎖 (Table-level lock)、頁鎖 (Page-level lock) 和行鎖 (Row-level lock，ASE 11.9.2 之後的版本才有提供此機制)。通常頁鎖比表鎖的限制更少 (或更小)，而行鎖又比頁鎖的限制更少 (或更小)。行鎖會對單行進行鎖定、頁鎖對本頁的所有行進行鎖定，而表鎖則鎖定整個表。此乃為了減小用戶間的資料爭用和改進衝突性，如下圖所示：



圖二：鎖的三種粒度 (資料來源：Sybase SGV2)

當 Sybase ASE 決定一個語句將查詢整個表或表的大多數頁時，它用表鎖來提供更有效的鎖定。鎖定策略直接受查詢方案約束，如果 update 或 delete 語句沒有可用的索引，它就執行表掃描或請求一個表鎖定。如果 update 或 delete 語句使用了索引，它就通過請求頁鎖來開始，如果影響到大多數行，它就要請求表鎖。一旦一個語句累積的頁鎖超過鎖提升值，Sybase ASE 就設法給該物件分配一個表鎖。如果成功了，頁鎖就不再必要了，因此被釋放。表鎖也在頁層提供避免鎖衝突的方法。

### ► 鎖的狀態

Sybase ASE 加鎖有三種狀態：

- 1) 意向鎖 (intend) -- 一種表級鎖，他表示在一個資料頁上獲得一個 S 或 X 鎖的意向。意向鎖可以防止其他交易該資料頁的表上獲得排它鎖。
- 2) 阻塞 (blocking, 簡記 blk) -- 表明目前加鎖程序的狀態，帶有 blk 尾碼的鎖說明該程序目前正阻塞另一個需要獲得鎖的程序，只有這一程序完成，其他程序才可以進行。
- 3) 需求鎖 (demand) -- 表示此時該程序企圖得到一個排它鎖。它可以防止在這一表或頁上加過多的 S 鎖，她表示某一事務是下一個去鎖定該表和該頁的交易。

需求鎖是一個內部過程，因此用 sp\_lock 是無法看見的。

## 二、鎖死 DEADLOCK

簡單地說，有兩個用戶程序，每個程序都在一個單獨的頁或表上有一個鎖，而且每個程序都想在對方程序的頁或表上請求不相容鎖時就會發生“鎖死”。在這種情況下，第一個程序在等待另一程序釋放鎖，但另一程序要等到第一個程序的物件釋放時才會釋放自己的鎖。

Sybase ASE 檢查是否鎖死，並終止交易中 CPU 時間累積最小的用戶（即最後進入的用戶）。Sybase ASE 會 Rollback 該用戶的交易，並用訊息 1205 通知有此鎖死行為的應用程式，然後允許其他用戶程序繼續進行。在多用戶情形下，每個用戶的應用程式都應檢查每個修改資料的交易是否有 1205 訊息，以此確定是否有可能鎖死。訊息 1025 表示該用戶的交易因鎖死而終止並被 Rollback。應用程式必須重新開始這個交易處理。

## 三、發現鎖死原因

既然管理資訊系統長時間鎖死的原因是由於我們提交或者是提交不當，那麼我們就可以透過修改程式防止出現鎖死。定位鎖死出錯處主要經過以下三步：

- 1) 在鎖死出現時，用 SP\_WHO, SP\_LOCK 獲得程序與鎖的活動情況。
- 2) 結合 sysobjects 和相應的操作員 (Login) 資訊表查出被鎖的資料表與鎖住別人的操作員。

3) 根據鎖定的資料表與操作員，可以估計出程式大約的出錯處。查詢操作員在鎖死時執行的具體操作即可完全定位出處錯處。最後找出程式並修改之。

#### ► 用 sp\_who 獲取被阻礙程序的資訊

系統過程 sp\_who 給出系統程序的報告。如果用戶的命令正被另一進程保持的鎖阻礙，則：

status 列顯示 “lock sleep”

blk 列顯示持有該鎖或這些鎖的程序標識，即被誰鎖定了。

loginame 列顯示登錄操作員。結合相應的操作員資訊表格，便可知道操作員是誰。

```

                                sp_who 11
  fid  spid  status      loginame  origname  hostname  blk  dbname  cmd
-----
  11    11  sleeping    diana     diana     olympus    0   sales   SELECT
  11    16  lock sleep    diana     diana     olympus   18   sales   WORKER PROCESS
  11    17  lock sleep    diana     diana     olympus   18   sales   WORKER PROCESS
  11    18  send sleep    diana     diana     olympus    0   sales   WORKER PROCESS
  11    19  lock sleep    diana     diana     olympus   18   sales   WORKER PROCESS
  11    20  lock sleep    diana     diana     olympus   18   sales   WORKER PROCESS
  11    21  lock sleep    diana     diana     olympus   18   sales   WORKER PROCESS

```

圖三：sp\_who 獲取被阻礙程序的資訊（資料來源：Sybase Perf）

#### ► 用 sp\_lock 瀏覽鎖

要得到關於當前 Sybase ASE 上持有的鎖的報告，可用系統程序 sp\_lock [spid1[,spid2]]，spid1,spid2 是表 master.dbo.sysprocesses 中的 server 程序 id，用 sp\_who 可以得到鎖定與被鎖定的 spid：

locktype 列顯示加鎖的類型和封鎖的粒度，有些鎖的尾碼還帶有 blk 表明鎖的狀態。首碼表明鎖的類型：Sh—共用鎖，Ex—排它鎖或更新鎖，中間表明鎖死在表上（“table” 或 ‘intent’）還是在頁上（page）。尾碼 “blk” 表明該程序正在阻礙另一個需要請求鎖的程序。一旦正在阻礙的程序一結束，其他程序就向前移動。“demand” 尾碼表明當前共用鎖一釋放，該程序就申請互斥鎖。

table\_id 列顯示表的 id，結合 sysobjects 即可查詢被封鎖的表名。

fid	spid	loid	locktype	sp_lock		page	row	dbname	context
				table_id					
0	15	30	Ex_intent	208003772		0	0	sales	Fam dur
0	15	30	Ex_page	208003772		2400	0	sales	Fam dur, Ind pg
0	15	30	Ex_page	208003772		2404	0	sales	Fam dur, Ind pg
0	15	30	Ex_page-blk	208003772		946	0	sales	Fam dur
0	30	60	Ex_intent	208003772		0	0	sales	Fam dur
0	30	60	Ex_page	208003772		997	0	sales	Fam dur
0	30	60	Ex_page	208003772		2405	0	sales	Fam dur, Ind pg
0	30	60	Ex_page	208003772		2406	0	sales	Fam dur, Ind pg
0	35	70	Sh_intent	16003088		0	0	sales	Fam dur
0	35	70	Sh_page	16003088		1096	0	sales	Fam dur, Inf key

圖四：sp\_lock 瀏覽鎖（資料來源：Sybase Perf）