

第十三章 資料鎖定

13-1 SQL Server的鎖定機制

13-1-1 鎖定對象

13-1-2 鎖定方法

13-1-3 鎖定的相容性

13-1-4 如何避免死結

13-2 鎖定的相關設計

13-1 SQL Server的鎖定機制

- 鎖定是 Microsoft SQL Server Database Engine 所使用的機制，用以同步處理多個使用者在同一時間存取相同資料。
- 在交易取得資料目前狀態的相依性前 (例如讀取或修改資料)，它必須保護自己使其免於受到另一個交易修改相同資料的影響。交易可以要求資料的鎖定以達到此目的。
- 鎖定有不同的模式，例如共用或獨佔。鎖定模式可定義交易在資料上的相依性層級。若已授與該資料的鎖定模式給某個交易，就不會再授與鎖定給另一個交易，以免造成衝突。如果交易所要求的鎖定模式，將和已授與相同資料的鎖定造成衝突，Database Engine 將停止要求交易，直到釋放第一個鎖定為止。
- 應用程式通常不會直接要求鎖定。Database Engine 中的鎖定管理員會在內部管理鎖定。
- Microsoft SQL Server Database Engine 擁有多個資料粒度的鎖定 (Lock)，允許交易鎖定不同類型的資源。爲了把鎖定的成本降至最低，Database Engine 自動依照工作的適當層級來鎖定資源。
 - 鎖定於較小的資料粒度 (Granularity) 如資料列可以提高並行，但如果鎖定許多的資料列則由於必須持有更多的鎖定而造成更高的負擔。
 - 鎖定於較大的資料粒度如資料表，從並行的角度來看則由於鎖定整個資料表會限制其他交易對於資料表其他部份的存取因而更費時。但由於必須維持的鎖定較少因此負擔較低。

13-1-1 鎖定對象

- 鎖定機制的原則是儘量鎖定於較小範圍的資料

資源	描述
1. RID (Row Identifier)	資料列識別碼，用來鎖定堆積內單一資料列。
2. KEY	索引中的資料列鎖定，用來保護可序列化交易中的索引鍵範圍。
3. PAGE	資料庫中的 8 KB 頁面，例如資料或索引頁面。
4. EXTENT	連續八個頁面的群組，例如資料頁或索引頁面。
5. HOBT	堆積(heap)或 B 樹狀(B-tree)目錄。針對沒有叢集索引的資料表中之資料頁，保護其索引或堆積的鎖定。
6. TABLE	一整個資料表，包含所有資料和索引。
7. FILE	資料庫檔案。
8. APPLICATION	應用程式指定資源。
9. METADATA	中繼資料鎖定。
10. ALLOCATION_UNIT	配置單位。
11. DATABASE	一整個資料庫。

13-1-2 鎖定方法

- SQL Server在不同時機，會使用不同鎖定方法

鎖定模式	說明
1.共用(Share) S	使用於不變更或更新資料的作業(唯讀作業)，例如 SELECT陳述式。
2.更新(Update) U	使用於可被更新的資源，防止多個工作階段正在讀取、鎖定、以及稍後可能更新資源時所發生的一般型式的死結(Deadlock)。
3.獨占(Exclusive) X	使用於資料修改動作，如 INSERT、UPDATE或 DELETE，確保不能對相同資源同時進行多重更新。
4.意圖(Intent)	建立鎖定階層架構，意圖鎖定又可分為： <ul style="list-style-type: none">•意圖共用(intent shared, IS)•意圖獨占(intent exclusive , IX)•意圖獨占共用(shared with intent exclusive, SIX)
5.結構描述(Schema)	當執行依存於資料表結構描述的作業時使用。結構描述鎖定的類型為：結構描述修改(Sch-M) 與結構描述固定性(Sch-S)。
6.大量更新 (BU)	當大量複製資料到資料表中並且指定 TABLOCK 提示時使用。
7.索引鍵範圍 (Key-range)	在交易中針對資料列範圍的鎖定，以防止其他交易在同一個範圍內新增、更改或刪除資料。

1. 共用鎖定

- 功能：允許並行交易在封閉式並行控制之下讀取 (SELECT) 資源。
- 當資源存在共用 (S) 鎖定時，任何交易都無法修改資料，除非交易隔離等級是設為可重複讀取或更高等級，或是使用鎖定提示來保持交易期間的共用 (S) 鎖定，否則讀取作業一完成就會釋放資源的共用 (S) 鎖定。

2. 更新鎖定

- 功能：可防止常見的死結。
- 在可重複讀取或可序列化交易中，交易在讀取資料時取得資源 (頁面或資料列) 的共用鎖定，然後修改資料，此過程需要將鎖定轉換為獨佔 (X) 鎖定。
- 如果兩筆交易取得某個資源的共用模式鎖定，然後嘗試同時更新資料，則其中一筆交易便會試著將鎖定轉換為獨佔 (X) 鎖定。這種「從共用模式到獨佔」的鎖定轉換必須等待，因為某一筆交易的獨佔鎖定與另一筆交易的共用模式鎖定並不相容，所以會發生鎖定等候。第二筆交易便嘗試取得更新時的獨佔 (X) 鎖定。由於兩筆交易都轉換成獨佔 (X) 鎖定，且兩者皆等候另一筆交易釋放其共用模式的鎖定，因此便發生死結。
- 為了避免這種潛在的死結問題，則使用更新 (U) 鎖定。一次只有一筆交易可以取得資源的更新 (U) 鎖定。交易如果修改資源，更新 (U) 鎖定便轉換為獨佔 (X) 鎖定。

3. 獨佔鎖定

- 功能：防止並行交易存取某個資源。
- 運用獨佔 (X) 鎖定，沒有其他交易可修改資料；只有使用 NOLOCK 提示或讀取未認可隔離等級，才能進行讀取作業。
- 資料修改陳述式 (例如 INSERT、UPDATE 和 DELETE) 結合了修改和讀取作業。陳述式先執行讀取作業來取得資料，再執行必要的修改作業。
- 資料修改陳述式通常會同時要求**共用鎖定和獨佔鎖定**。例如，UPDATE 陳述式可能基於與一個資料表的聯結來修改另一個資料表的資料列。
 - 對已更新的資料列要求**獨佔鎖定**
 - UPDATE 陳述式還對聯結資料表中讀取的資料列要求**共用鎖定**

4. 意圖鎖定

- 意圖鎖定將共用鎖定或獨佔鎖定放在**鎖定階層中較低的資源上**。
- 意圖鎖定有兩個用途：
 - 防止其他交易修改較高層級的資源，而導致較低層級的鎖定失效。
 - 爲了改進 Database Engine 偵測資料粒度較高層級的鎖定衝突的效率。
- 例如：
 - 資料表內的頁面或資料列要求**共用鎖定**之前，先在**資料表層級**要求共用意圖鎖定。在資料表層級上設定意圖鎖定讓另一筆交易無法後續取得包含該分頁之資料表的獨佔鎖定。
 - 意圖鎖定可以提昇效能，因爲 Database Engine 只會在資料表層級上檢查意圖鎖定，來判斷交易是否可安全地取得該資料表的鎖定。這種方式省略了必須檢查資料表的每個資料列或分頁的鎖定來判斷交易是否可以鎖定整個資料表的需求。
- 意圖鎖定包括意圖共用 (IS)、意圖獨佔 (IX) 與共用意圖獨佔 (SIX)。

鎖定模式	描述
意圖共用 (IS)	<ul style="list-style-type: none"> 保護在階層較低位置的某些（但不是全部）資源上要求的或取得的共用鎖定。
意圖排除 (IX)	<ul style="list-style-type: none"> 保護在階層較低位置的某些（但不是全部）資源上要求的或取得的獨佔鎖定。IX 是 IS 的超集，它也保護在較低層級資源要求的共用鎖定。
意圖排除共用 (SIX)	<ul style="list-style-type: none"> 保護對階層較低位置的所有資源要求的或取得的共用鎖定，以及對某些（但非全部）較低層級資源的意圖排除鎖定。 在最上層的資源中允許同時發生的 IS 鎖定。例如，在資料表上取得 SIX 鎖定也會取得所修改頁面的意圖排除鎖定，以及所修改資料列的獨佔鎖定。每個資源一次只能有一個 SIX 鎖定以防止其他的交易更新資源，雖然其他的交易可藉由取得資料表層級的 IS 鎖定來讀取階層架構中位於較低層級的資源。

註：另有意圖更新 (IU)、共用意圖更新 (SIU)、更新意圖排除 (UIX)等，參考線上說明

5. 結構描述鎖定

- 結構描述修改 (Schema Modification, **Sch-M**) 鎖定：
 - 在執行資料表的資料定義語言 (DDL) 作業 (例如加入資料行或卸除資料表) 時使用。
 - 在結構描述修改鎖定保留期間，禁止資料表的並行存取。這表示結構描述修改鎖定以外的所有作業將遭到封鎖，直到釋放鎖定為止。
- 結構描述固定性 (Schema Stability, **Sch-S**) 鎖定：
 - 在編譯查詢時使用。
 - 結構描述固定性鎖定並未封鎖任何交易式鎖定，包括獨佔 (X) 鎖定。
 - 因此其他的交易在查詢編譯期間可以繼續執行，包括獨佔 (X) 鎖定資料表的交易。但是 DDL 動作無法執行於資料表。
- The Database Engine uses schema stability (Sch-S) locks when compiling and executing queries. Sch-S locks do not block any transactional locks, including exclusive (X) locks. Therefore, other transactions, including those with X locks on a table, continue to run while a query is being compiled. However, concurrent DDL operations, and concurrent DML operations that acquire Sch-M locks, cannot be performed on the table.

6. 大量更新鎖定

- 大量更新 (Bulk Update, BU) 鎖定是在大量複製資料到資料表以及已：
 - 指定 TABLOCK 提示(TABLOCK :將資料表所採用的共用鎖定保留到陳述式結束為止。如果也指定了 HOLDLOCK，就會將共用資料表鎖定保留到交易結束為止)。
 - 或在 sp_tableoption 設定並啓用 table lock on bulk load 資料表選項時使用。
- 大量更新 (BU) 鎖定允許多個執行緒將資料同時大量載入到相同資料表，同時禁止未大量載入資料的其他處理序存取該資料表。

7. 索引鍵範圍鎖定

- 使用可序列化交易隔離等級時，保護 Transact-SQL 陳述式所讀取之記錄集內隱含包括的資料列範圍。
- 索引鍵範圍鎖定可防止幽靈讀取。透過保護資料列之間的索引鍵範圍，也可防止某交易存取的記錄集內的幽靈插入或刪除。

13-1-3 鎖定的相容性

- 只有相容的鎖定類型才能放在欲鎖定的資源上
 - 如**獨占(X)鎖定**時，其他的交易皆**無法**取得該資源的任何一種鎖定
 - 如資源已套用**共用(S)鎖定**，則即使第一筆交易尚未完成，其他的交易仍可取得該項目的**共用鎖定**或**更新(U)鎖定**，故共用及更新鎖定是相容的

鎖定模式的相容性

	(1). 現有已授與的模式					
(2). 要求的模式	IS	S	U	IX	SIX	X
意圖共用 (IS)	是	是	是	是	是	否
共用 (S)	是	是	是	否	否	否
更新 (U)	是	是	否	否	否	否
意圖獨占 (IX)	是	否	否	是	否	否
與意圖獨占共用 (SIX)	是	否	否	否	否	否
獨占 (X)	否	否	否	否	否	否

13-1-4 如何避免死結 (Deadlock)

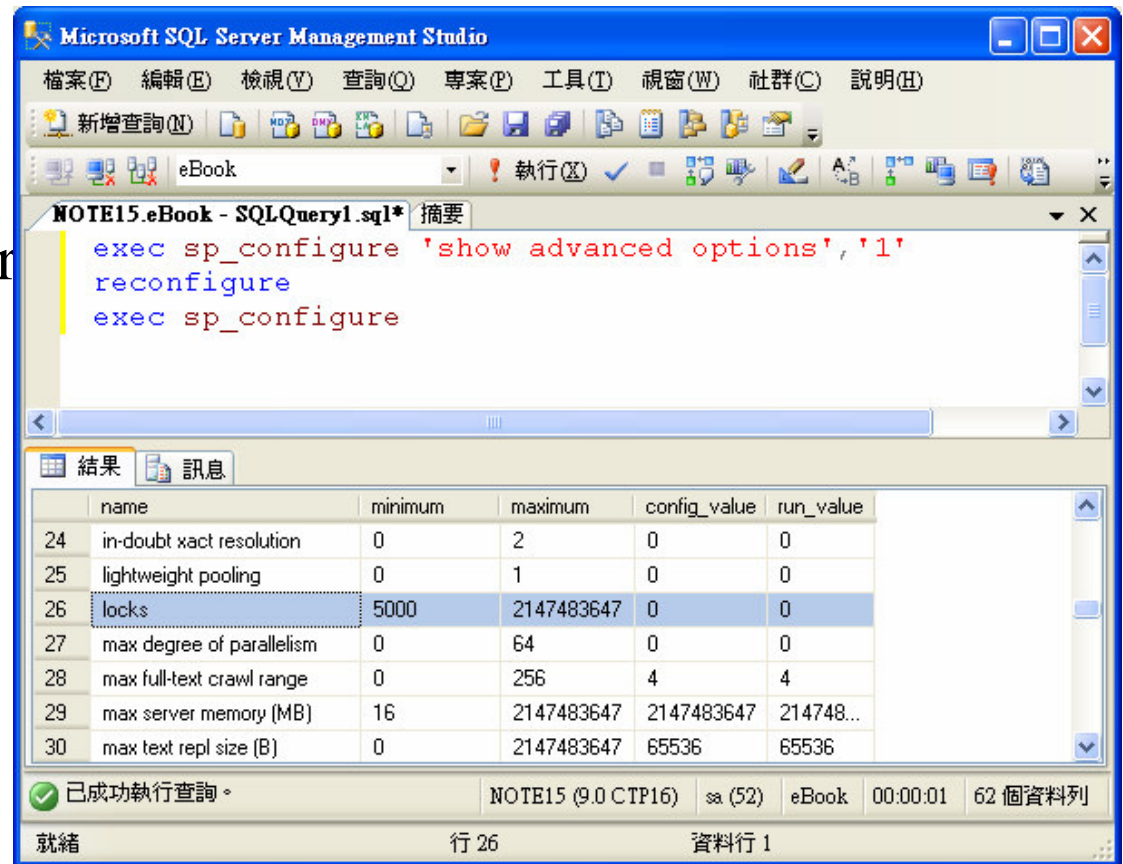
- 死結之意是兩個以上的執行緒或執行動作，對於資源的鎖定及釋放，具有循環性的依存關係：
 - 如執行緒A的處理動作依序需要鎖定資料表1及2，另一執行緒B依序需要鎖定資料表2及1，兩個執行緒都在等待對方釋放資源，此時即為死結

如何避免死結

- 以相同的順序來存取物件
- 在交易中避免使用者互動
- 將交易維持在單一批次中且愈短愈好
- 使用低隔離等級
- 使用繫結連線

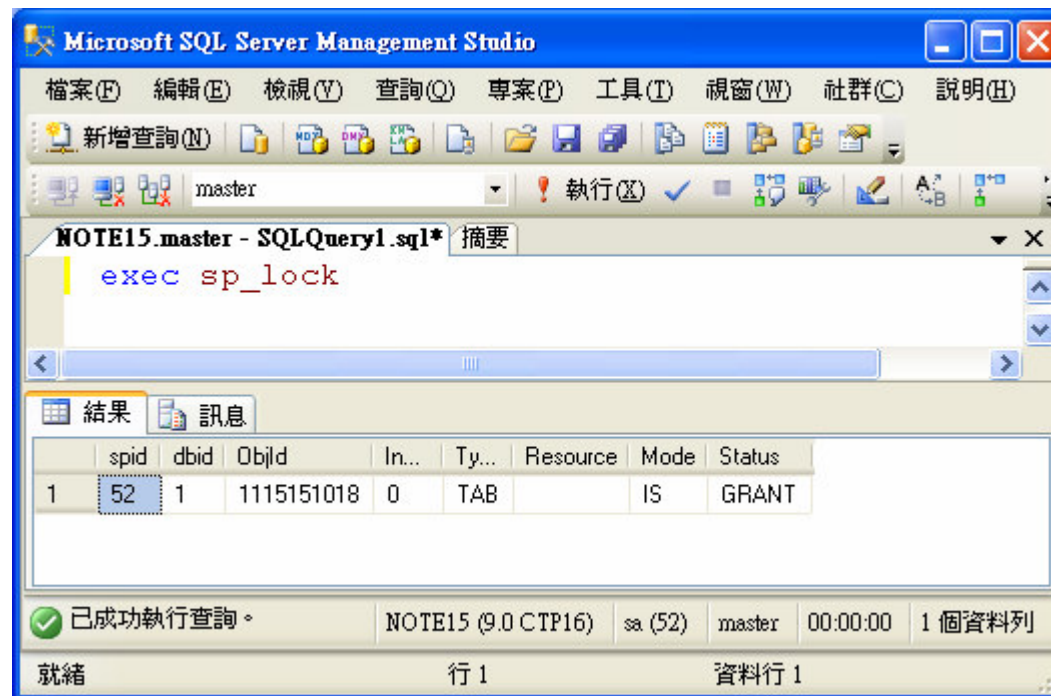
13-2 鎖定的相關設計

- sp_configure的
Locks
 - 設定鎖定數量上限，預設值是0，表示由SQL Server依記憶體大小，動態配置鎖定數量



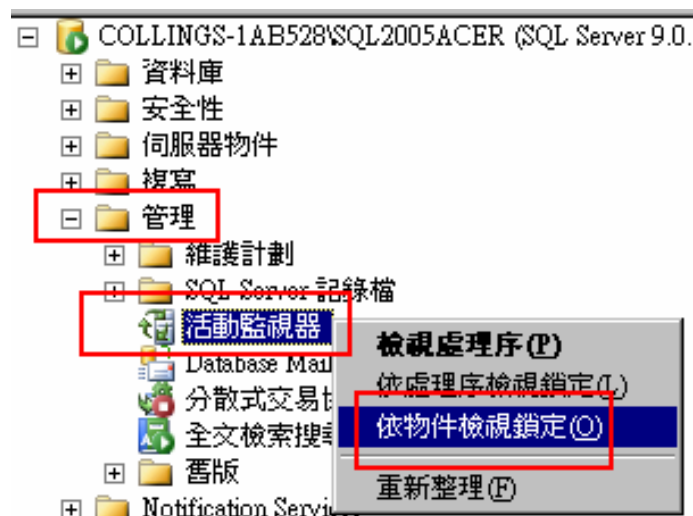
鎖定的相關設計

- sp_lock
 - 查看關於鎖定的資訊



鎖定的相關設計

- 檢視鎖定-使用活動監視器



處理序識別碼	內容	批次識別碼	類型	子類型	物件識別碼	描述	要求模式	要求類型	要求狀態	擁有者類型	擁有者識別碼	擁有者 ...	資料庫	物件
51	0	0	DATABASE		0		S	LOCK	GRANT	SHARED_TRANSA...	0	0000000...	msdb	(內部)
52	0	0	DATABASE		0		S	LOCK	GRANT	SHARED_TRANSA...	0	0000000...	ReportServer	(內部)
55	0	0	OBJECT		13		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	OBJECT		15		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	HOB		720575940...		Sch-M	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	OBJECT		26		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	OBJECT		34		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	OBJECT		41		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	OBJECT		54		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	HOB		562949956...		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	METADATA	INDE...	0	object...	Sch-S	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	METADATA	DAT...	0	data_s...	Sch-S	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	OBJECT		717245610		Sch-M	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	OBJECT		5		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	OBJECT		4		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)
55	0	0	OBJECT		7		IX	LOCK	GRANT	TRANSACTION	59540	0000000...	tempdb	(內部)