**Full name : Nguyễn Cảnh Thương-HE163742**

**Class : IA1604**

**Subject:DBI202**

**Transaction Isolation Levels**

Transaction Isolation Levels là thước đo mức độ thành công của cô lập giao dịch. Cụ thể, mức độ cô lập của giao dịch được xác định bằng sự có mặt hoặc không có các hiện tượng sau:

* **Dirty Reads** : Tác động này xảy ra khi 1 transaction A đọc 1 row khi nó đang được update bởi 1 transaction B khác và chưa được commit.Transaction A sẽ đọc dữ liệu vẫn chưa được commit.

 Ví dụ : 2 Transaction T3, T4 thực hiện đồng thời, T3 rút $20, T4 gửi thêm $200 nhưng lại hủy bỏ sau đó.

Giả sử, Người dùng A và Người dùng B đang truy cập vào một hàng trong bảng cùng một lúc. Người dùng A muốn đọc và Người dùng B muốn cập nhật hàng. Trong trường hợp chênh lệch múi giờ, các giao dịch được thực hiện. Vì vậy, khi Người dùng B chưa cập nhật hàng (trong quá trình cập nhật), Người dùng A đọc hàng đó và nhận được bản ghi cũ có thể không đúng với hoạt động của họ. Tình trạng này được gọi là Dirty Read.

* **Non-Repeatable Problem: Trường hợp này xảy ra khi 1 transaction A đọc 1 đơn vị dữ liệu nhiều lần và kết quả khác nhau giữa các lần đo giữa thời gian đọc của các lần đó , dữ liệu bị 1 transaction khác commit thay đổi**

Đây còn được gọi là vấn đề không lặp lại.Giả sử người dùng A select một hàng , cột hoặc bảng để đọc .Cùng lúc đó người dùng B lại chỉnh sửa như là thêm , sửa , xóa . Rồi người dùng A lại select lần thứ 2 thì sẽ có 2 kết quả khác nhau . Đó là sự không nhất quán trong dữ liệu.

Ví dụ : Giao tác T5 và T6 cùng thực hiện đồng thời, T5 rút tiền, T6 tính tổng số dư của 3 tài khoản x,y,z. Khi kết thúc 2 giao tác, kết quả T6 không chính xác.

* **Phantoms**  : Là tình trạng mà một giao tác đang thao tác trên một tập dữ liệu nhưng giao tác khác lại chèn thêm các dòng dữ liệu vào tập dữ liệu mà giao tác kia quan tâm.Trường hợp này xảy ra khi mà Transaction A cập nhật 1 dữ liệu ở hàng này và transaction B cập nhật dữ liệu ở hàng khác thì đó transaction A reset lại sẽ nhận được một thông tin mới .

|  |  |  |  |
| --- | --- | --- | --- |
| Isolation Level | Dirty Read | Non Repeatable Read | Phantom |
| Read uncommitted | Yes | Yes | Yes |
| Read commited | No | Yes | Yes |
| Repeatable read | No | No | No |
| Serializable | No | No | No |

Bây giờ, chúng ta sẽ đi qua tất cả  năm Isolation levels và tìm ra giải pháp cho tình huống của chúng ta :

**+ Read Uncommitted – đọc dữ liệu chưa chuyển giao :**

Đặc điểm:

* + Không thiết lập Shared Lock trên những đơn vị dữ liệu cần đọc. Do đó không phải chờ khi đọc dữ liệu (kể cả khi dữ liệu đang bị lock bởi giao tác khác)
  + Vẫn tạo Exclusive Lock trên đơn vị dữ liệu được ghi, Exclusive Lock được giữ cho đến hết giao tác

Ưu điểm

* + Tốc độ xử lý rất nhanh
  + Không cản trở những giao tác khác thực hiện việc cập nhật dữ liệu

Nhược điểm

* + Có khả năng xảy ra mọi vấn đề khi xử lý đồng thời :

**+ Read Commited**

Đặc điểm:

* + Đây là mức độ cô lập mặc định của SQL Server
  + Tạo Shared Lock trên đơn vị dữ liệu được đọc, Shared Lock được giải phóng ngay sau khi đọc xong dữ liệu
  + Tạo Exclusive Lock trên đơn vị dữ liệu được ghi, Exclusive Lock được giữ cho đến hết giao tác

Ưu điểm:

* + Giải quyết vấn đề Dirty Reads.
  + Shared Lock được giải phóng ngay, không cần phải giữ cho đến hết giao tác nên không cản trở nhiều đến thao tác cập nhật của các giao tác khác.

Nhược điểm:

* + Chưa giải quyết được vấn đề Unrepeatable Reads, Phantoms,Lost Updates
  + Phải chờ nếu đơn vị dữ liệu cần đọc đang được giữ khoá ghi (xlock)

**+ Repeatable Read**

Đặc điểm

* + Tạo Shared Lock trên đơn vị dữ liệu được đọc và giữ shared lock này đến hết giao tác => Các giao tác khác phải chờ đến khi giao tác này kết thúc nếu muốn cập nhật, thay đổi giá trị trên đơn vị dữ liệu này .
  + (Repeatable Read = Read Committed + Giải quyết Unrepeatable Reads).
  + Tạo Exclusive Lock trên đơn vị dữ liệu được ghi, Exclusive Lock được giữ cho đến hết giao tác.

Ưu điểm

* + Giải quyết vấn đề Dirty Reads và Unrepeatable Reads

Nhược điểm

* + Chưa giải quyết được vấn đề Phantoms, do vẫn cho phép insert những dòng dữ liệu thỏa điều kiện thiết lập shared lock
  + Phải chờ nếu đơn vị dữ liệu cần đọc đang được giữ khoá ghi (xlock)
  + Shared lock được giữ đến hết giao tác ==> cản trở việc cập nhật dữ liệu của các giao tác khác

+ **Serializable**

Đặc điểm

* + Tạo Shared Lock trên đơn vị dữ liệu được đọc và giữ shared lock này đến hết giao tác => Các giao tác khác phải chờ đến khi giao tác này kết thúc nếu muốn cập nhật, thay đổi giá trị trên đơn vị dữ liệu này .
  + Không cho phép Insert những dòng dữ liệu thỏa mãn điều kiện thiết lập Shared Lock (sử dụng Key Range Lock) ==> Serializable = Repeatable Read + Giải quyết Phantoms
  + Tạo Exclusive Lock trên đơn vị dữ liệu được ghi, Exclusive Lock được giữ cho đến hết giao tác.

Ưu điểm

* + Giải quyết thêm được vấn đề Phantoms

Nhược điểm

* + Phải chờ nếu đơn vị dữ liệu cần đọc đang được giữ khoá ghi (xlock)
  + Cản trở nhiều đến việc cập nhật dữ liệu của các giao tác khác