**Môn học: Quản trị hệ cơ sở dữ liệu**

**Bài thực hành:**

**GIÁM SÁT HỆ THỐNG**

**Nội dung (Lý thuyết ở chương 8):**

**Tham khảo: Chapter 10 – Sách Beginning SQL Server 2008 Administration**

**BÁO CÁO NỘI DUNG THỰC HÀNH**

**Họ tên sinh viên:**

**Mã số sinh viên:**

**Lớp:**

**Số máy tính:**

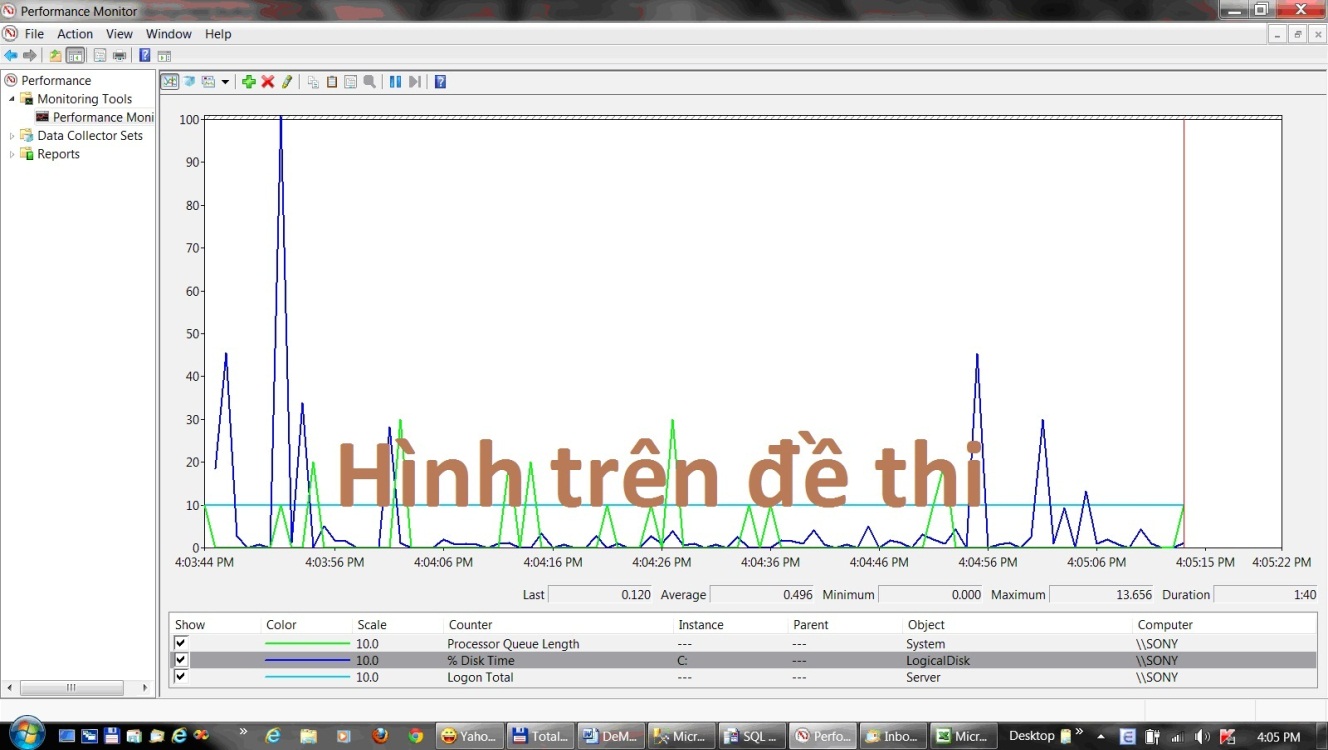
**Ngày, buổi thực hành:**

**Các câu hỏi, bài thực hành: Sinh viên có thể trả lời bằng cách gõ văn bản hay chụp màn hình dán vào ngay tại sau mỗi câu hỏi:**

**(Bài thực hành này thực hiện trên bản developer trở lên)**

**PERFORMANCE MONITOR**

1. Khởi động Performance monitor (Start🡪 Run, gõ “Perfmon”; hay mở Performance monitor từ công cụ Profiler)
2. Từ công cụ Performance monitor mở cửa sổ “Resource monitor”, tìm hiểu nội dung và chụp các tab: Overview, CPU, Memory, Disk, Network.
3. Sử dụng công cụ Performance monitor để giám sát hoạt động hiện hành của hệ thống bằng sơ đồ dạng line của các Counter sau: (lần lượt chụp mỗi hình chứa 2 hay 3 counter có màu khác nhau và có tỉ lệ trên thước đo phù hợp dể quan sát), ví dụ

****

* 1. Các Processor counter:
     1. *% Processor Time*
     2. *% Processor Time (sqlservr)*
     3. *Processor Queue Length*
  2. Các Disk counter:
     1. *Avg. Disk Queue Length*
     2. *% Disk Time*
  3. Các Memory counter:
     1. *Memory:Pages/Sec*
     2. *Memory:Available Bytes*
     3. *Process:Working Set (sqlservr)*
     4. *SQL Server: Buffer Manager: Buffer Cache Hit Ratio*
     5. *SQL Server: Buffer Manager: Page Life Expectancy*
  4. Network counter*: Network Interface: Bytes Total/Sec*
  5. Các SQL Server counter:
     1. *SQL Server: General Statistics: User Connections*
     2. *SQL Server: Locks: Average Wait Time*
     3. *SQL Server: Locks: Deadlocks/Sec*
     4. *SQL Server Access Methods: Page Splits/sec*

**DYNAMIC MANAGEMENT VIEWS**

1. Select dữ liệu các counter từ system view: ***sys.dm\_os\_performance\_counters***

**LOG FILE VIEWER**

1. Dùng Log file viewer để xem log file của SQL server và log file của hệ điều hành

**ACTIVITY MONITOR**

1. Dùng Activity monitor để xem ***Process info***
2. Dùng Activity monitor để xem***Locks by Process***
3. Dùng Activity monitor để xem***Locks by Object***

**SYSTEM STORED PROCEDURES**

1. Dùng system stored procedure **sp\_who** để xem thông tin các process
2. Dùng system stored procedure **sp\_who2** để xem thông tin các process
3. Tạo tình huống có block và hủy process gây ra block bằng lệnh **KILL** theo bài thực hành sau (trong sách Beginning): ***Try it Out-System Stored Procedures***

Try it Out-System Stored Procedures

Take a look at what information is returned by the system stored procedures, and how you can use them to isolate troublesome processes.

First, open a query window. Type and execute the following code:

USE AdventureWorks;

GO

BEGIN TRAN

UPDATE Person.Contact

SET LastName = 'Gates'

WHERE ContactID = 1;

Open a second query window. Type and execute the following code:

USE AdventureWorks;

GO

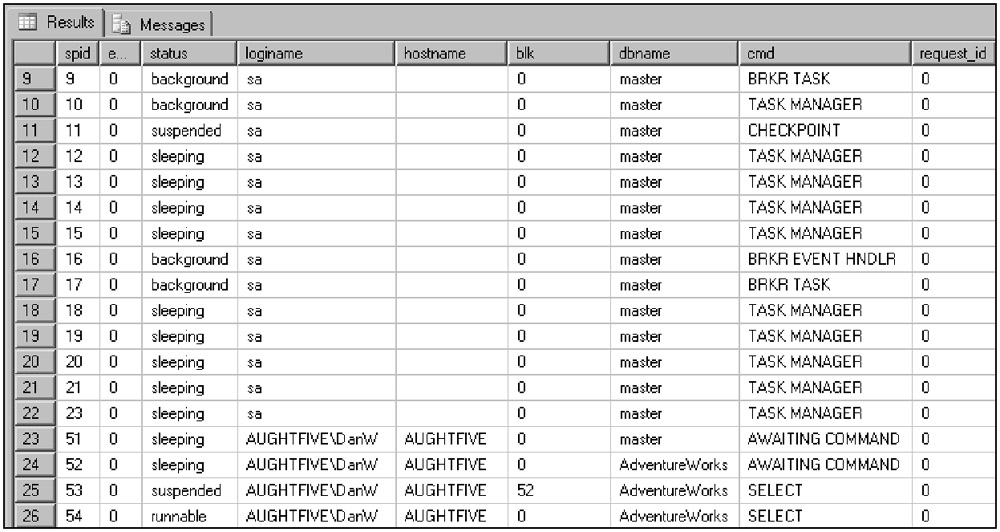
SELECT \* FROM Person.Contact

WHERE ContactID = 1;

Now, open a third query window and run the sp\_who system stored procedure by executing the following command:

EXEC sp\_who;

Notice that one of the processes shows that it is being blocked by another session. In the case shown in [Figure 10-7](mk:@MSITStore:D:\Teaching\QTHCSDL\QTHCSDL-HK2-0910\TaiLieu\beginning-sql-server-2005-administration.9780470047040.27703.chm::/final/BBL0077.html#ch10fig07#ch10fig07), SPID 53 is being blocked by SPID 52.

  
Figure 10-7: Result of running sp\_who system stored procedure

Now, execute the sp\_who2 stored procedure, but restrict the result set to the Server Process ID (SPID) that is responsible for the block in progress. In my case, the spid is 52.

EXEC sp\_who2 52;

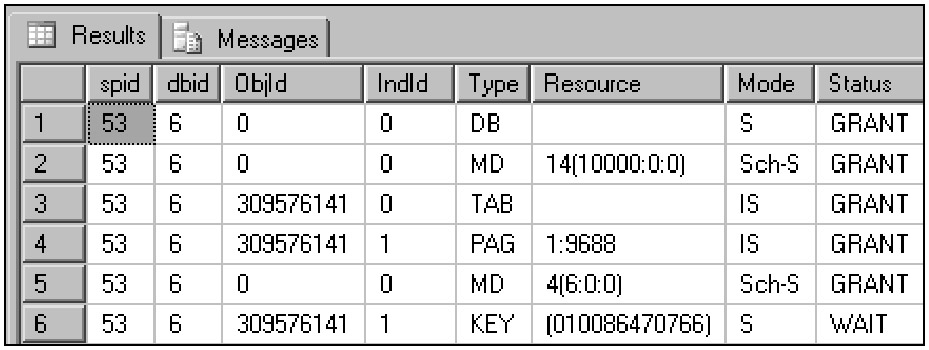
The more comprehensive results of the sp\_who2 stored procedure execution return very useful information (such as the program and user responsible, as well as when the session executed the command responsible for the lock contention).

Next, identify what object is being contested by the two processes. Execute the sp\_lock stored procedure. The results of this procedure, like the sp\_who and sp\_who2 stored procedures, can be restricted by passing in the appropriate process ID.

Type and execute the following command to display the information about the SPID being blocked. This is the SPID that returned a value in the BlkBy column of the sp\_who2 results. For me, it was 53, but remember that your SPID will most likely be different:

EXEC sp\_lock 53;

The results are shown in [Figure 10-8](mk:@MSITStore:D:\Teaching\QTHCSDL\QTHCSDL-HK2-0910\TaiLieu\beginning-sql-server-2005-administration.9780470047040.27703.chm::/final/BBL0077.html#ch10fig08#ch10fig08).

  
Figure 10-8: sp\_lock results

In [Figure 10-8](mk:@MSITStore:D:\Teaching\QTHCSDL\QTHCSDL-HK2-0910\TaiLieu\beginning-sql-server-2005-administration.9780470047040.27703.chm::/final/BBL0077.html#ch10fig08#ch10fig08), notice that several locks have been requested and granted, but the shared lock on the clustered index key 010086470766 (which represents the contact in the Person.Contact table with the ID of 1) is in a WAIT status. This is because spid 52 is currently modifying that particular row and has an exclusive lock on that key.

To terminate the blocking process, execute the KILL command specifying the appropriate SPID, which for me, is 52:

KILL 52;

*Use caution when killing a process.* SPID 52 *is the process on my computer. Your results may vary!*

**PROFILER**

1. Tạo mới một trace đặt tên Trace1, chọn hiện tất cả event và tất cả các column. Chọn giám sát các events: Objects: created; Audit login và Audit logout. Khi trace đang chạy (runing), mở SQL Server management studio để đăng nhập, tạo mới 1 CSDL, đăng xuất. Xem và báo cáo các dòng thông tin trong trace tương ứng với các sự kiện đã thực hiện.
2. **Analyzing Deadlocks with Profiler:** Đọc, hiểu và thực hành theo sách Beginning: ***Try it Out-Analyzing Deadlocks with Profiler***

#### Detect and Analyze Long Running Queries with Profiler: Đọc, hiểu và thực hành theo sách Beginning: *Try it Out-Analyzing Queries*

**DATABASE TUNING ADVISOR (DTA)**

1. Tạo mới 1 session, chọn workload là file MySQL.sql. Chọn CSDL là Adventureworks, bảng Production.Product. Dùng DTA tiến hành phân tích CSDL khi chạy lệnh trong file sql và cho biết kết quả DTA đề nghị.
2. Đọc, hiểu và thực hành theo sách Beginning: ***Try it Out-Using the DTA to Analyze a SQL Script***

**USING THE DTA WITH PROFILER**

1. Đọc, hiểu và thực hành theo sách Beginning: ***Try it Out-Correlating Performance Data with Profiler Data***