

37회 mcpworld SQL 고급과정 세미나

Microsoft SQL Server 2005 대기 및 큐 진단분석

필라넷, SQL Server 아카데미
성대중 책임

djsung@feelanet.com

blog.naver.com/dreamania_73

강사 소개



성대중

Email: djsung@feelanet.com

Blog: blog.naver.com/dreamania_73

Community:

http://www.dbguide.net/club/sub.jsp?club_id=sqldna

근무 이력

현) 필라넷 / DB 사업부 / 책임 컨설턴트

전) 영림원 소프트랩 ERP 컨설턴트 / 개발자

주요 업무

SQL Server 컨설팅 / 기술지원 / 개발 / 교육

강사 활동

SQL Server 아카데미, TechNet 웹캐스트, HandsOnLab 등

출판물

SQL Server 2005 관리자 가이드 (Microsoft Korea)

SQL Server 2005 개발자 가이드 (Microsoft Korea)

SQL Server 2005 포켓 컨설턴트 관리자용(정보문화사)

inside SQL Server 2005 T-SQL Programming(정보문화사)

Database Refactoring (위키북스)



목차

- Mission 1 : 문제해결의 단서를 찾아라!
 - SQL Server 2005 진단분석 및 성능튜닝 방법론
 - 성능모니터를 사용한 SQL Server 2005 종합진단
- Mission 2 : 과학적 접근방법으로 범인을 잡아라!
 - SQL Server 2005 대기 유형
 - 대기 및 큐 분석
- Mission 3 : 프로토콜로 시스템화하자!
 - DMVStats 의 소개
 - DMVStats 의 활용방안

Mission1: 문제해결의 단서를 찾아라

- SQL Server 2005 운영환경 진단분석 방법론
- 문제의 정의(Mission 1)
- 모니터링 및 성능 베이스라인의 중요성
- 성능 모니터
 - 모니터링 대상 성능카운터(시스템)
 - 모니터링 대상 성능타운터(SQLServer)
- 성능 카운터간의 상호연관성 분석
- DEMO
- 문제의 해결



SQL Server 2005 운영환경 진단분석

진단분석 & 성능개선

주기적이고 체계적인 진단분석 & 성능개선 프로세스 수행 필요!

진단분석 절차

1. 데이터베이스 환경 모니터링
2. 병목 부분 리소스의 정밀진단
3. 문제 원인이 되는 개체 검색
4. 문제 원인 개체 성능 튜닝
5. 성능개선절차 시스템화

성능개선 절차

5. 서버 구성 최적화
4. 잠금 및 차단이슈 최적화
3. 인덱스 전략 최적화
2. 쿼리 최적화
1. 스키마 디자인

문제의 정의(Mission1)

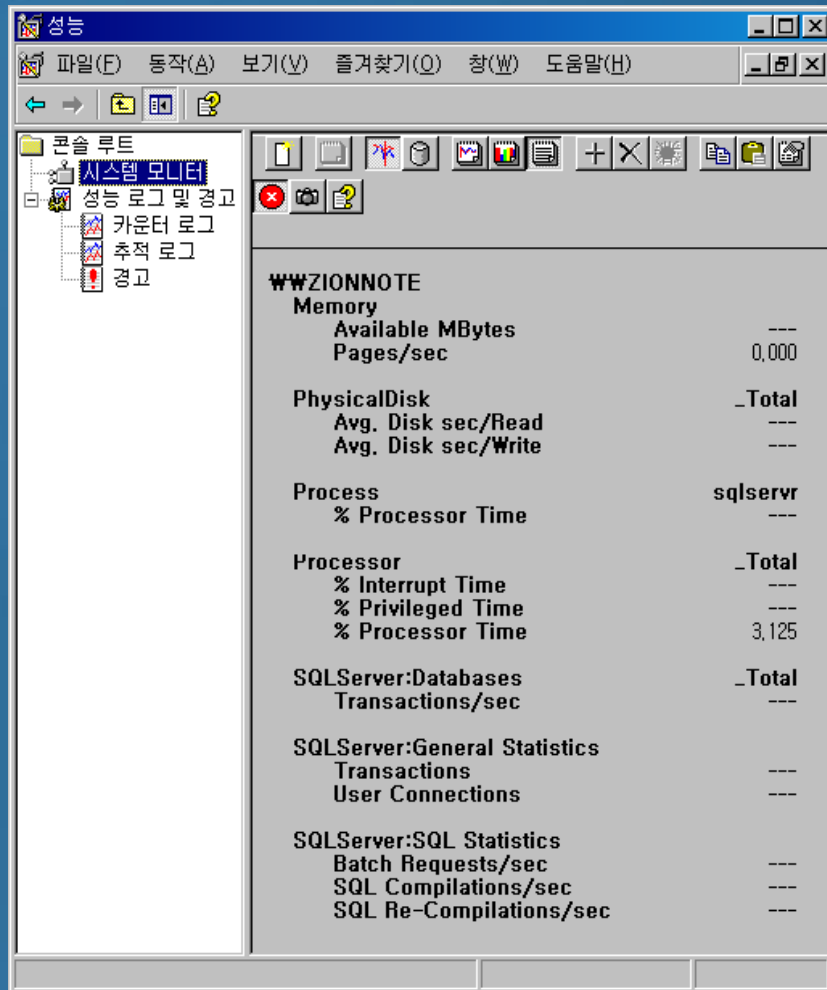
- 사건발생 : 데이터베이스 서버의 성능저하
- 사용자
 - 서버가 느려졌어요! 업무를 볼 수가 없어요!
 - 원래는 이렇지 않았는데 갑자기 이상하게 느려요!
- 운영자
 - 도대체 오늘은 왜 이런거지?
 - 휴~~ 오늘도 집에 일찍 가는 것은 포기해야겠네 OTL
- 관리자
 - 도대체 정확한 원인이 뭐야? 언제까지 해결할 수 있는 거야?
 - 이번엔 또 뭐라고 보고 드리지?

모니터링과 베이스라인의 중요성

- 모니터링
 - VITAL SIGN
- 성능 베이스라인
 - 판단 기준이 되는 시점의 시스템의 평균 성능 지표로 구성
 - 추세(Trend), 이력관리, 여러 시점의 베이스라인 필요(기준, 결산...)
- 적용절차
 - 얼마나 느려졌나? 원래는 얼마 정도였나?
 - 베이스라인과 달라진 것이 무엇이 있나?
 - 문제의 원인은 무엇이고, 그에 대한 대응방안은 무엇인가?
 - 문제해결을 위해 소요예측시간은 얼마인가?
 - 문제해결 후 보고를 위한 데이터는 이력관리되는가?

성능 모니터(PERFMON.EXE)

전반적인 성능 정보를 제공해 주는 종합검진 도구



- 실시간 모니터링
 - 현재 서버상태 모니터링
- 성능 로그
 - 추세분석, 이력관리
 - 성능 카운터 간의 연관관계 분석
- 적용절차
 - 평상시 모니터링을 위한 표준 성능카운터 템플릿
 - 특정 리소스에 대한 병목현상 발생여부 진단을 위한 템플릿
 - 성능 정보를 요약보고하기 위한 보고서용 템플릿

모니터링 대상 성능카운터(시스템)

시스템 리소스에 대한 성능 정보를 수집

- CPU
 - PROCESSOR:%Processor Time
 - PROCESS:%Processor Time(sqlservr)
 - PROCESSOR:%User Time, Privileged Time
 - SYSTEM:Processor Queue Length
 - SYSTEM:Context Swithes/sec
- MEMORY(vs. SQL Server Buffer Manager)
 - Available MBytes
 - Pages/sec, Page Faults/sec
- DISK
 - Avg. sec/Read, Avg. sec/Write
 - Avg. Disk Queue Length

모니터링 대상 성능카운터(SQLServer)

SQL Server 관련 리소스에 대한 성능 정보를 수집(1/3)

- Workload
 - PROCESSOR:%Processor Time
 - PROCESS:%Processor Time(sqlservr)
 - PROCESSOR:%User Time, Privileged Time
 - SYSTEM:Processor Queue Length
 - SYSTEM:Context Swithes/sec
- Lock & Latch
 - Available MBytes
 - Pages/sec, Page Faults/sec

모니터링 대상 성능카운터(SQLServer)

SQL Server 관련 리소스에 대한 성능 정보를 수집(2/3)

- Workload
 - SQLStatistics:Batch Requests/sec
 - SQLStatistics:SQL Compilations/sec
 - SQLStatistics:SQL Re-compilations/sec
 - General Statistics:User Connections
 - Databases:Transactions/sec
- Lock & Latch
 - Locks:Average Lock Wait Time(ms)
 - Locks:Lock Requests/sec
 - Locks:Lock Waits/sec
 - Latches:Average Latch Wait Time(ms)
 - Latches:Latch Waits /sec

모니터링 대상 성능카운터(SQLServer)

SQL Server 관련 리소스에 대한 성능 정보를 수집(3/3)

- Access Methods
 - Access Methods:Index Searches/sec
 - Access Methods:Full Scans/sec
 - Access Methods:Forward Records/sec
 - Access Methods:Page Splits/sec
- Buffer Mgr. & Plan Cache & Memory Mgr.
 - Buffer Manager:Buffer Hit Ratio
 - Buffer Manager:Page Life Expectancy
 - Buffer Manager:Checkpoint/sec
 - Buffer Manager:Lazy writers/sec
 - Plan Cache:Cache Hit Ratio
 - Memory Mgr:Memory Grant Pending

성능카운터간의 상호연관성 분석

성능카운터를 상호비교함으로써 새로운 성능정보 제공

- CPU 이슈
 - Kernal Mode vs. User Mode
 - Processor:%Processor Time vs. Process :%Processor Time(sqlservr)
- 컴파일 및 재컴파일 이슈
 - Batch Requests/sec vs. SQL Compilations/sec
 - SQL Compilations/sec vs. SQL Re-compilations/sec
- 디스크 이슈
 - Disk Queue Length vs. Disk sec/Transfer
- 메모리 이슈
 - Page Life Expectancy, Checkpoints/sec, LazyWrites/sec

데모

성능 모니터 사용법(optional)

성능 모니터 분석 사례

- 엑셀
- SQL Tuning Manger(가칭, Feelanet DB 사업부)
- Spotlight(3rd party 도구, Quest Software)

문제해결(Mission 1)

- 방법 : 성능 모니터를 활용한 전체 시스템 성능 진단
- 확인내역(예제)
 - CPU
 - 일 평균 사용률은 20%이나, Peak Time(오전9-10, 오후6-8)에는 평균 80%
 - 평균 사용률은 80%이상, 그 중 SQL Server가 72%를 사용
 - 메모리
 - Buffer Hit Ratio가 급격히 떨어지는 현상 발생
 - Page Life Expectancy가 평균 300미만으로 메모리부족
 - 디스크
 - 트랜잭션 로그 파일이 위치한 E 드라이브의 Avg. sec/Read, Avg. sec/Write 값이 200 ms 으로 높게 나타남
 - SQL Server
 - 전체 Batch Requests/sec 중 약 80%가 초기 컴파일 발생
 - Full Scans/sec 이 초당 100 으로 매우 높게 나타남

Mission2: 과학적으로 범인을 잡아라

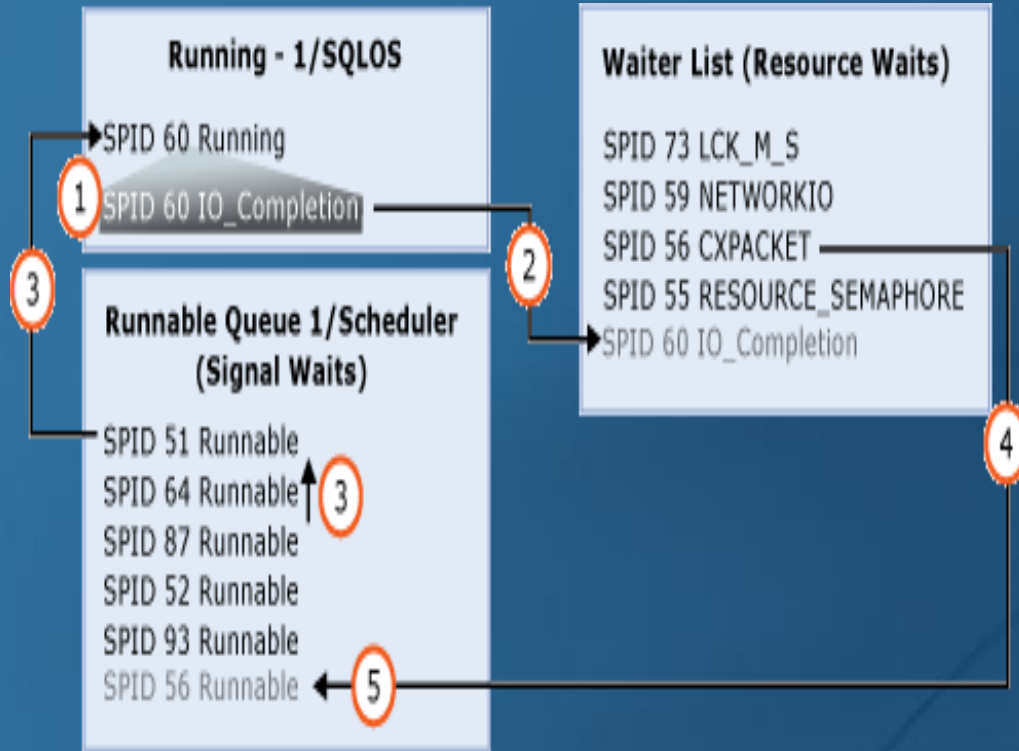
- SQL Server 2005 실행모델
- SQL Server 2005 실행모델 대기시간
- 대기자 리스트와 대기 유형
- 대기 분석을 위한 DMV, DMF, 스크립트
- OLTP 환경에서의 성능 및 튜닝 청사진
- 리소스 병목현상의 유형 및 대책
- 알아두어야 할 대기유형 및 개선방안
- DEMO



문제의 정의(Mission2)

- 사건발생 : 데이터베이스 차단이슈 발생
- 사용자
 - 평소에는 빨리 실행되는데, 간혹 너무 오래 걸려요!
- 운영자
 - 서버에 CPU 사용률은 그리 높지 않은데 뭐가 문제지?
 - 어? 내가 테스트할 때는 이상없이 실행되었는데...
 - 죄송합니다. 서버에 잠깐 문제가 있었나 봐요 OTL
- 관리자
 - 서버를 업그레이드해야 하나?

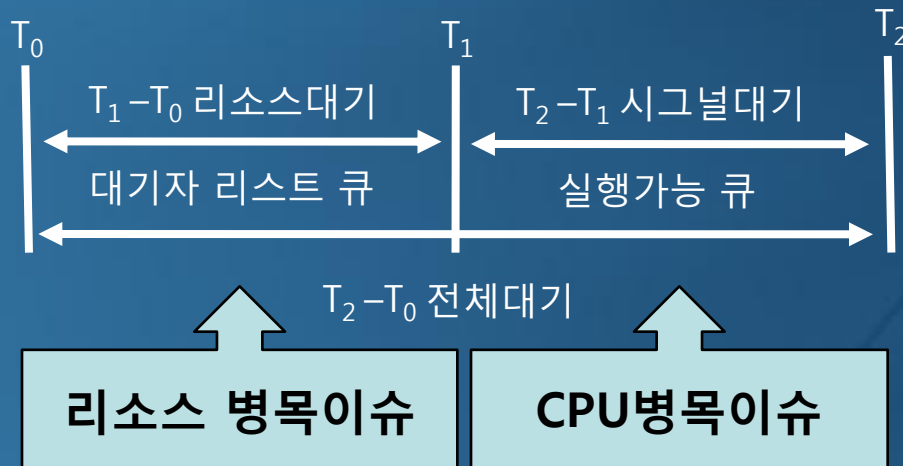
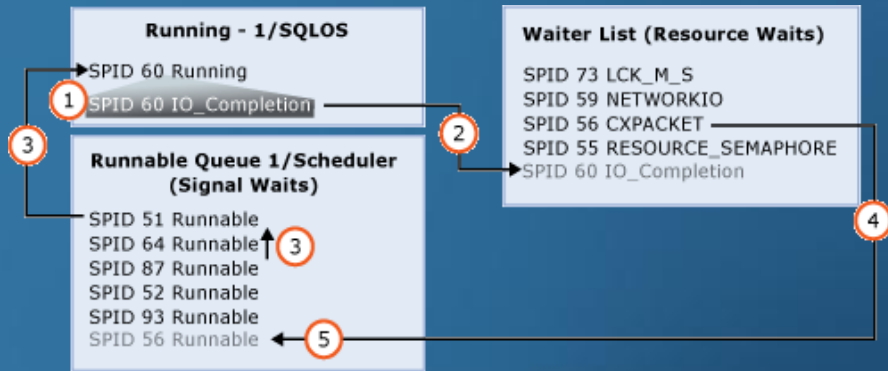
SQL Server 2005 실행모델



• 실행단계

- ① SPID 60 물리적 IO가 필요한 페이지 요청
- ② SPID 60 대기자 리스트로 이동
- ③ SPID 51 실행중 상태로 이동
- ④ SPID 64~93 1단계씩 위로 이동
- ⑤ SPID 56 병렬처리 완료 후 실행가능한 큐 마지막으로 이동

SQL Server 2005 실행모델 대기시간



• 대기시간

- T_0 : 리소스를 요청, 리소스 대기 시작
- T_1 : 리소스 사용가능시점, 실행가능 큐에서 CPU대기
- $T_1 - T_0$: 리소스 대기시간
- T_2 : CPU 사용가능시점, 실행 중 상태로 변경
- $T_2 - T_1$: 시그널대기 시간
- $T_2 - T_0$: 전체 대기 시간

대기자 리스트와 대기유형

- 대기자 리스트(waiter list)
 - 대기자 리스트로 이동할 때 세션에 대기유형 부여
 - sys.dm_os_waiting_tasks DMV
 - 대기시간 : wait_duration_ms
 - 시그널 대기시간 : signal_wait_time_ms
 - 차단주체 : blocking_session_id
 - 차단리소스 정보 : resource_description
 - 리소스 대기시간
 - $\text{Wait_duration_ms} - \text{signal_wait_time_ms}$ 로 계산
- 실행가능(Runnable) 큐
 - sys.dm_exec_requests DMV 중 State가 Runnable 인 세션

대기분석을 위한 DMV, DMF, 스크립트

- DMV,DMF
 - sys.dm_exec_requests
 - Status = {runnable, running, suspended, sleeping,...}
 - sys.dm_os_waiting_tasks
 - WaitType, blocking_session_id, resource_description
 - sys.dm_os_wait_stats
 - DBCC SQLPERF('sys.dm_os_wait_stats',CLEAR)
- 저장 프로시저 <http://www.microsoft.com/technet/scriptcenter/scripts/sql/sql2005>
 - track_waitstats_2005
 - get_waitstats_2005
 - get_statements_in_waiter_list

OLTP 환경을 위한 성능 청사진(1/7)

OLTP 환경과 OLAP 환경 비교

OLTP	비교항목	OLAP
비교적 많음	트랜잭션의 양	비교적 적음
읽기/쓰기 혼재	트랜잭션의 종류	대부분 읽기전용
비교적 소용량쿼리	쿼리의 부하	대용량쿼리
거의 필요없음	병렬처리	필요에 따라 사용
적극 권장	실행계획재사용	개별 쿼리 최적화 필요
꼭 필요한 것만(최소화)	인덱스의 수	필요하다면 추가(최대화)
병행성 중심	튜닝접근방법	개별쿼리최적화 중심

OLTP 환경을 위한 성능 청사진(2/7)

OLTP 환경에서 피해야 할 시나리오

- 데이터베이스 설계 이슈

설명	값	원천정보
여러 개의 테이블이 조인되는 쿼리가 자주 반복 실행되는 경우	>4	Sys.dm_exec_sql_text Sys.dm_exec_cached_plans
여러 개의 인덱스가 설정된 테이블에 자주 업데이트가 발생하는 경우	>3	Sys.indexes sys.dm_db_operational_index_stats
대량 IO가 발생하는 쿼리, 테이블 스캔, 범위 스캔이 발생하는 쿼리	>1	성능 모니터 개체 SQL Server Access Methods Sys.dm_exec_query_stats
사용되지 않는 인덱스	Sys.dm_db_index_usage_stats DMV에 나타나지 않는 인덱스. 인덱스가 전혀 사용되지 않으면, sys.dm_db_index_usage_stats DMV에 나타나지 않습니다.	

OLTP 환경을 위한 성능 청사진(3/7)

OLTP 환경에서 피해야 할 시나리오

- CPU 이슈

설명	값	원천정보
시그널 대기	>25%	Sys.dm_os_wait_stats
쿼리 실행계획 재사용	<90%	성능 모니터 개체 SQL Server Statistics
병렬처리: Cxpacket 대기	>5%	Sys.dm_os_wait_stats

OLTP 환경을 위한 성능 청사진(4/7)

OLTP 환경에서 피해야 할 시나리오

- 메모리 이슈

설명	값	원천정보
Page life expectancy	<300 초	성능 모니터 개체 SQL Server Buffer Manager SQL Server Buffer Nodes
Page life expectancy	50%이하로 떨어 짐	Perfmon object SQL Server Buffer Manager
Memory Grants Pending	>1	성능 모니터 개체 SQL Server Memory Manager
SQL cache hit ratio	<90%	SQL cache hit ratio 성능 카운터 수치가 90% 이하로 떨어져 60초 이상 지속되는 경우.

OLTP 환경을 위한 성능 청사진(5/7)

OLTP 환경에서 피해야 할 시나리오

• IO 이슈

설명	값	원천정보
Average Disk sec/read	>20 ms	성능 모니터 개체 Physical Disk
Average Disk sec/write	>20 ms	성능 모니터 개체 Physical Disk
대용량 IO, 테이블 스캔, 범위 스캔을 발생시키는 쿼리	>1	성능 모니터 개체 SQL Server Access Methods
상위 2위의 대기 유형에 다음과 같은 대기유형이 포함되는 경우: ASYNCH_IO_COMPLETION, IO_COMPLETION, LOGMGR, WRITELOG, PAGEIOLATCH_x	상위 2 위	Sys.dm_os_wait_stats
초당 읽기/쓰기 바이트 수가 낮은 경우		성능 모니터 개체 Physical Disk

OLTP 환경을 위한 성능 청사진(6/7)

OLTP 환경에서 피해야 할 시나리오

• 차단이슈

설명	값	원천정보
차단 발생 퍼센트	>2%	Sys.dm_db_index_operational_stats
Block process report	30 초	Sp_configure 저장 프로시저, SQL Server 2005 프로파일러
Average Row Lock Waits	>100ms	Sys.dm_db_index_operational_stats
상위 2위의 대기유형에 다음과 같은 유형의 대기유형이 포함되는 경우: LCK_*	상위 2위	Sys.dm_os_wait_stats
데드락이 다수 발생하는 경우	시간당 >5	추적플래그 1204를 사용하여 에러로그에 데드락 정보를 기록하거나, 프로파일러를 사용하여 deadlock graph 이벤트를 수집

OLTP 환경을 위한 성능 청사진(7/7)

OLTP 환경에서 피해야 할 시나리오

- 네트워크 이슈

설명	값	원천정보
데이터베이스로의 라운드 트립이 자주 발생하면서, 높은 네트워크 지연현상이 있는 경우	Output queue length >2	성능모니터 개체: Network Interface
네트워크 대역폭이 거의 대부분 사용되는 경우	Packets Outbound Discarded Packets Outbound Errors Packets Received Discarded Packets Received Errors	성능모니터 개체: Network Interface

리소스 병목현상의 유형 및 대책

리소스 병목현상

- 리소스 대기
 - 대기자 리스트에 머무는 시간
 - 세션에서 요청하는 리소스가 가용한 상태까지 대기
- 리소스 대기의 진단
 - 대기 및 큐 분석을 통하여 진단 가능
 - 관련 스크립트

<http://www.microsoft.com/technet/scriptcenter/scripts/sql/sql2005/default.mspix>

리소스 병목현상의 유형 및 대책

리소스 병목현상 이슈의 유형

- 리소스 병목현상 관련 이슈
 - 메모리 압박 및 IO 서브시스템
 - IO 대기
 - 인덱스의 누락 및 잘못된 설정된 인덱스
 - 대량 IO를 발생시키는 쿼리
 - 쿼리 실행계획 재사용
 - 차단
- 대기자 리스트에 있는 SQL 문 조회
 - get_statements_in_waiter_list 저장 프로시저

데모

DMV를 이용한 리소스 병목현상의 진단

- sys.dm_io_virtual_file_stats
- sys.dm_db_missing_index_*
- sys.dm_exec_query_stats
- sys.dm_exec_cached_plans
- sys.dm_db_index_operational_stats

SQL Server 2005 프로파일러로 차단진단(Optional)

알아두어야 할 대기유형 및 개선방안

대기유형의 진단

- 진단절차
 - sys.dm_os_wait_stats의 누적값을 기준 상위 5위 점검
 - 누적값 기준 상위 5위의 대기유형이 발생하는 원인 찾기
 - 쿼리 조사
 - 인덱스 조사
 - 리소스 조사
 - 해결방안 도출 및 적용
 - 해결방안 적용 후 대기유형에 대한 진단분석
 - DBCC SQLPERF('sys.dm_os_wait_stats', CLEAR)
- 내부전용, 시스템발생 대기유형은 분석대상에서 제외

알아두어야 할 대기유형 및 개선방안

대기유형 (1/4)

- OLEDB

- 원인 : OLE DB 공급자 호출
(4 Part Name 호출, 원격 프로시저 호출, OPENQUERY,
OPENROWSET, DMV 호출, 프로파일러 추적)
- 개선방안 : OLE DB를 사용한 어플리케이션, 쿼리 튜닝

- CXPACKET

- 원인 : 병렬처리로 인한 대기
- 개선방안 : 병렬처리를 발생시키는 쿼리 튜닝, 누락된
인덱스 추가

알아두어야 할 대기유형 및 개선방안

대기유형(2/4)

- PAGEIOLATCH_*, PAGELATCH_*

- 원인 :

- PAGEIOLATCH_* : 데이터페이지의 IO가 완료할 때까지 대기
 - PAGELATCH_* : 동일 페이지 INSERT 경합, 자동증가, Page Split 등

- 해결방안

- 메모리 증설, 쿼리튜닝, 인덱스 추가, 디스크증설, 데이터 파일 추가

- WRITELOG, IO_COMPLETION

- 원인 : 비-데이터페이지(트랜잭션 로그 등) 의 IO가 완료할 때까지 대기

- 해결방안 : IO의 분산, IO 대역폭의 추가, 트랜잭션 로그 드라이브 분리

알아두어야 할 대기유형 및 개선방안

대기유형(3/4)

- LCK_*

- 원인 : 다른 세션의 의해 개체(행,페이지,테이블)의 잠금설정
- 추가진단 : 성능모니터 Avg. Lock Wait Time(ms) 참조
- 해결방안 : 공유잠금에 대해서는 트랜잭션 격리수준 조정, 트랜잭션의 지속시간 최소화

- Latch_*

- 버퍼 페이지를 제외한 나머지 내부 캐시의 경합해결을 위한 동기화 개체
- 원인 : 메모리 압박, 추가진단 : Avg. Latch Wait time(ms)

알아두어야 할 대기유형 및 개선방안

대기유형(4/4)

- CMEMTHREAD

- 원인 : 동시에 여러 개의 쿼리 실행하여, 메모리 할당 곤란
예) 동시에 수많은 Adhoc 쿼리가 플랜캐시에 등록
- 해결방안 : AdHoc 쿼리의 프로시저화

- RESOURCE_SEMAPHORE

- 원인 : 쿼리를 실행하기 위한 메모리 부족
- 추가진단 : 성능모니터의 Memory Grant Pending 카운터
- 해결방안 : 메모리 증설, 대량 메모리 요청 작업(해싱, 정렬 등) 쿼리 튜닝

데모

DMV를 이용한 대기유형의 진단

- sys.dm_os_wait_stats
- track_waitstats_2005
- get_waitstats_2005
- get_statements_in_wait_list

문제해결(Mission 2)

- 방법 : 차단 진단용 스크립트와 SQL 프로파일러를 활용
- 확인내역(예제)
 - 현재 OLE DB를 제외한 전체 대기유형 중 CXPACKET 전체 20%를 점유
 - 병렬처리가 발생할만큼 비용이 높은 쿼리 발생 의심
 - Full Scans/sec 가 50 으로 높게 나타남
 - 프로파일러로 추적한 결과, OOO 테이블에서 테이블 스캔 발생
 - IO 기준 상위 50위를 확인한 결과 대량 IO 발생 쿼리 발견
 - 누락된 인덱스를 찾기 위해 dm_db_missing_* 참조하여 인덱스 추가
 - USP_xxx 프로시저로 인해 OOO 테이블에 차단현상 발생
 - LCK_* 대기유형이 높게 나타남
 - 프로파일러의 Block Process Report로 차단발생한 쿼리 추적

Mission3: 프로토콜로 시스템화하자!

- 문제의 정의
- DMVStats 소개
- DMVStats 설치 및 구성
- DMVStats 분석 및 보고
- Demo
- 문제의 해결



문제의 정의(Mission 3)

- 사건발생 : 시스템 재시작 후 성능정보 사라짐
- DMV의 한계
 - 서버가 재시작되면 모든 정보가 초기화됨(Dynamic)
 - 현재 시점(현재까지의 누적)의 성능정보만 제공
 - 이력 및 추세에 대한 정보 필요
 - 사용하기 어렵다
- 사용자 요구사항
 - 이력 데이터(2주전, 1개월전 성능 정보)
 - 추세분석
 - 일시적인 대기 현상에 대한 정보 기록
 - 어플리케이션 변경 전후 성능 비교

DMVStats 란?

대기 및 큐 분석 성능 튜닝 방법론을 시스템화

- SQL Server 2005 Service Pack 2
 - 현 시점의 시스템 상태에 대한 사용자관점의 접근 가능
 - SQL Server Management Studio Custom Report 기능 지원
 - SQL Server 2005 Performance Dashboard 제공
- DMVStats 성능 데이터웨어하우스
 - DMV 정보를 주기적으로 수집
 - 추세 및 이력에 대한 보고서 기능 지원(리포팅 서비스)
 - 웹사이트 및 다운로드(ver 1.01, 2007-07-09)
<http://www.codeplex.com/sqldmvstats/>
- SQL Server Customer Advisory Team의 노하우 전수!

DMVStats 설치 및 주요기능

- SQL Server 2005 이후 버전에 대해서만 지원
- 주요 기능
 - DMV 데이터 수집
 - SQL 에이전트 자동화 작업으로 동작
 - DMV 데이터웨어하우스 저장소(Repository)
 - DMVStatsDB 데이터베이스로 추가됨
 - 분석 및 보고
 - SQL Server 2005 Reporting Service 보고서 기능 활용
- 설치
 - InstallDMVStatsDBProcs.cmd : DB, 테이블, SP 생성
 - DeployReports.cmd : Reporting Service 용 보고서 배포

DMVStats 구성(1/2)

Configure DMVstats

DMVstats SQL Agent Job Status: Enabled [disable jobs](#)

Databases Included

id	name	
8	AdventureWorks	remove
7	DMVstatsDB	remove
11	Northwind	remove
10	pubs	remove

Databases Not included

id	name	
1	master	add
3	model	add
4	msdb	add
9	NWReport	add
5	ReportServer	add
6	ReportServerTempDB	add
2	tempdb	add

- 모니터링 대상 DB를 선택
- DMVstats 데이터 수집 SQL 에이전트 작업 활성화
 - 기본값은 비활성화
- DMV 스냅샷 수집 간격설정
 - sys.dm_exec_query_stats
sys.dm_os_wait_stats
 - exec msdb.dbo.sp_update_schedule
@name=N'DMVstats Snapshot Schedule',
@freq_subday_interval=10
- DMV 샘플링 간격설정
 - sys.dm_exec_requests and
sys.dm_os_waiting_tasks
 - exec msdb.dbo.sp_update_schedule
@name=N'DMVstats Sample
Schedule',@freq_subday_type =
2,@freq_subday_interval = 30

DMVStats 구성(2/2)

비교기준 시점이 되는 성능 베이스라인을 설정

- 베이스라인의 설정
 - 시계열 분석
 - 어플리케이션 변경 전후
 - 추세 분석

Starting snapshot: 93: May 11 2007 9:00AM Ending snapshot: 105: May 11 2007 10:00AM View Report

Baseline name: BL2

1 of 1 100% Find Next Select a format Export

Configure Baselines

To create a new baseline, fill out the parameters above and click "View Report"

ID	Date	Start	End	Seconds	Name	Snapshot source
1	5/11/2007	1:10:15 AM	2:00:01 AM	2,986	BL1	Snapshots 1 through 9 delete
2	5/11/2007	9:00:00 AM	10:00:00 AM	3,600	BL2	Snapshots 93 through 105 delete

	S21		
	S20	SS5 13:05	Friday01pm BL
	S19		
	S18		
	S17		
	S16		
	S15	SS4 13:00	Friday11am BL
	S14		
	S13		
	S12		
	S11		
	S10	SS3 11:10	Friday11am BL
	S9		
	S8		
	S7		
	S6		
	S5	SS 2 11:05	Friday11am BL
	S4		
	S3		
	S2		
	S1		
Samples	Snapshots	Baselines	

DMVStats 수집데이터(1/2)

DMV의 기준 시점 스냅샷 정보

- DMV Snapshot

설명	테이블명	관련 성능이슈
데이터베이스 파일 IO 정보	virtual_file_stats	데이터베이스 IO 적체 여부
인덱스 통계정보	index_operational_stats	차단 및 핫스팟(hotspot) 여부
쿼리 통계정보	query_stats	최소, 최대, 평균 작업자 수, 경과 시간, IO 정보
대기 통계정보	wait_stats	리소스 대기과 CPU 대기 정보(성능카운터 정보와 상호 비교)
성능모니터 카운터	performance_counters	리소스 사용량과 가용성 정보
관리	instrumentation	각 DMV 정보 수집에 대한 기록

DMVStats 수집데이터(2/2)

DMV의 기준 시점 샘플링 정보

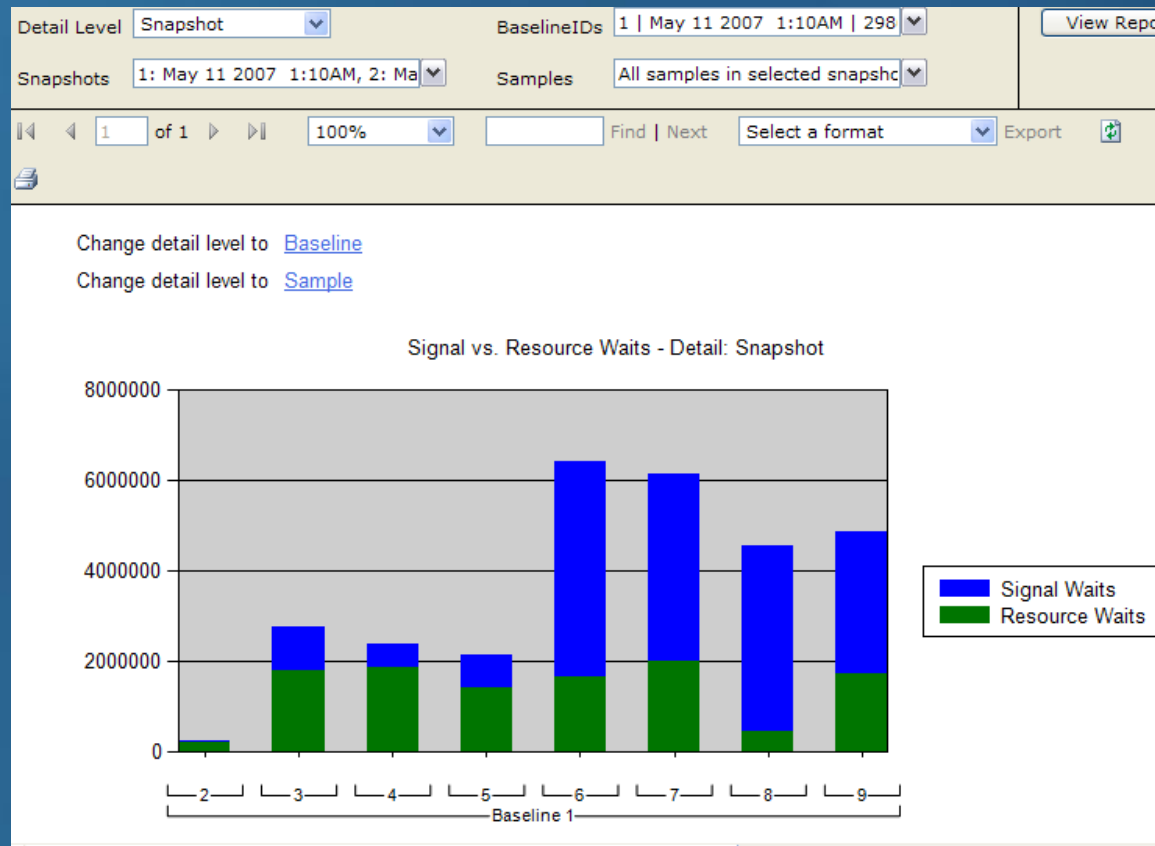
- DMV Samples

설명	테이블명	관련 성능 이슈
사용자 요청 및 대기자 리스트	requests_and_waiters	sys.dm_exec_requests DMV의 현재 사용자 요청 정보와 sys.dm_os_waiting_tasks DMV의 대기자 리스트 정보 수집
차단	block_info	차단이 발생한 SQL 문
스케줄러	Schedulers	최소, 최대, 평균 작업자 수, 경과시간, IO 정보
대기 통계정보	wait_stats	리소스 대기와 CPU 대기 정보(성능카운터 정보와 상호 비교)
성능 카운터 정보	performance_counters	리소스 사용량과 가용성 정보
관리	Instrumentation	각 DMV 정보 수집에 대한 기록

DMVStats 활용법

분석 및 보고

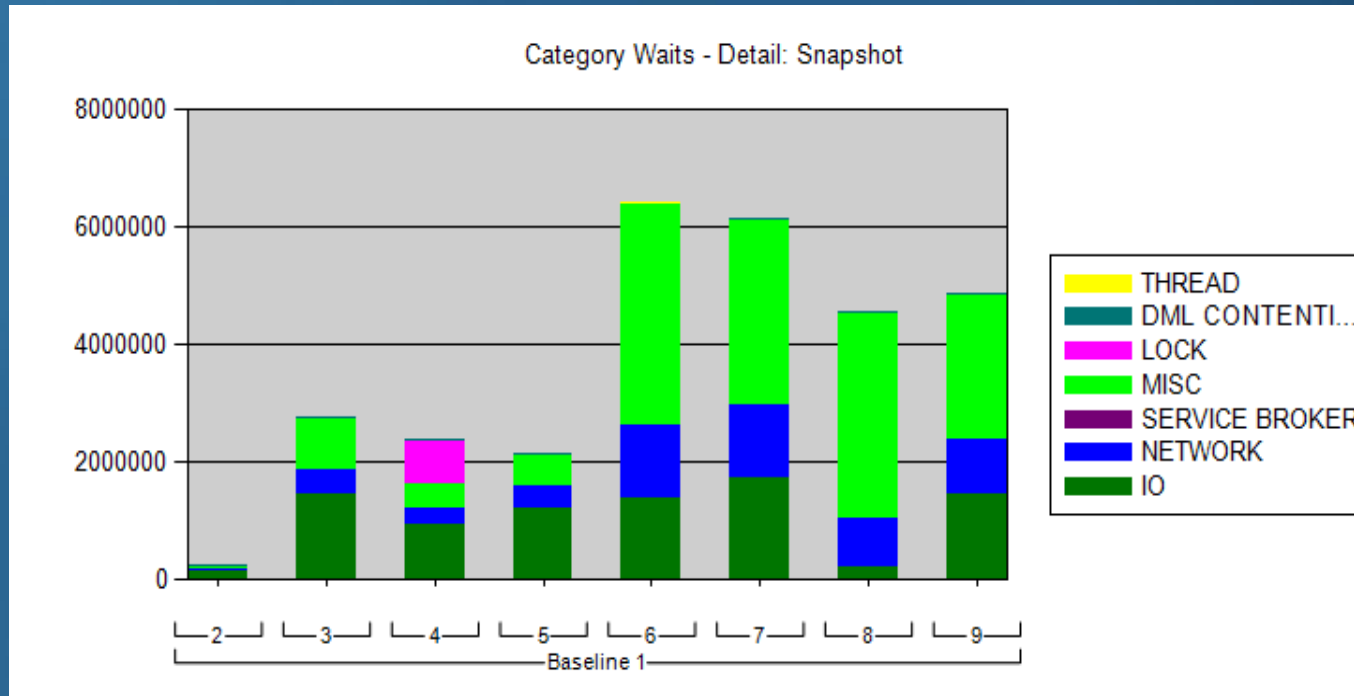
- 대기 통계 정보(리소스 대기 vs. 시그널 대기)



DMVStats 활용법

분석 및 보고

- 대기 통계 정보(각 대기유형별 점유율)



DMVStats 활용법

분석 및 보고

- 잠금 관련 대기유형 상세 정보 > 차단 정보



Category / Action	Report	Guidance
LOCK		
Drill into LOCK waits	Wait Stats	
Find biggest blocks (currently running) in the waiter list	Analyze Block Info	Display statements involved in the block
Find Index Contention	Index Operational Stats	Drill to snapshots, samples



Analyze Block Info

dbname	resource type	resource associated entity id	request mode	wait ms	Baseline ID	Snapshot ID
Northwind	PAGE	282888923447296	S	154812	1	4
waiting	SELECT C.CategoryName, ProductName, Yr=Substring(Convert(nvarchar(4), O.OrderDate, 111), 1, 4), TotalPurchase=ROUND(SUM(CONVERT(decimal(14,2), OD.Quantity * (1-OD.Discount) * OD.UnitPrice)), 0) FROM [Order Details] OD, Orders O, Products P, Categories C WHERE OD.OrderID = O.OrderID AND OD.ProductID = P.ProductID AND P.CategoryID = C.CategoryID GROUP BY C.CategoryName, ProductName, Substring(Convert(nvarchar(4), O.OrderDate, 111), 1, 4) ORDER BY C.CategoryName, ProductName, Substring(Convert(nvarchar(4), O.OrderDate, 111), 1, 4)					
blocking	create procedure SalesByYear AS -- This is provided "AS IS" with no warranties, and confers no rights. -- Use of included script samples are subject to the terms specified at http://www.microsoft.com/info/copyright.htm -- SELECT Year=DATEPART(yy, ShippedDate), Orders.OrderID, [Order Subtotals].Subtotal FROM Orders INNER JOIN [Order Subtotals] ON Orders.OrderID = [Order Subtotals].OrderID order by DATEPART(yy, ShippedDate), Orders.OrderID compute sum([Order Subtotals].Subtotal) by DATEPART(yy, ShippedDate)					
Northwind	PAGE	282888923447296	S	139720	1	4
Northwind	PAGE	282888923447296	S	110624	1	4
Northwind	PAGE	282888923447296	S	107844	1	4



데모

Performance Dashboard 사용법(Optional) DMVStats의 사용법 및 분석사례

- 시그널 대기 vs. 리소스 대기
- 대기 범주별 누적값 비교
- 차단 분석

문제해결(Mission 3)

- 방법 : DMVStats 를 통해 성능정보의 이력, 추세 분석 가능
- 확인내역(예제)
 - 사라지지 않는 성능 정보
 - 서비스 재시작 이후에도 성능 이력 정보 그대로 보관
 - 베이스라인 확보 및 다양한 분석 보고
 - 성능 베이스라인을 필요에 따라 정의하고 시계열 분석 가능
 - 다양한 분석과 보고서 제공
 - 퇴근 후 발생한 성능 저하의 원인은 OOO 테이블에 의한 차단이 원인
 - 이력 데이터에 대한 분석
 - 최근 1 개월간 사용자 증가 추세, IO 증가 추세
 - 1개월간 수집한 데이터를 스냅샷을 기준으로 추세 분석 가능
 - 하드웨어 용량 계획 및 향후 업그레이드 계획의 근거 자료로 활용 가능

참고자료

- SQL Server Best Practice WebSite
 - <http://technet.microsoft.com/ko-kr/sqlserver/bb331794.aspx>
- Technical White Paper
 - SQL Server 2005 Performance Tuning using Waits and Queues
http://www.microsoft.com/technet/prodtechnol/sql/bestpractice/performance_tuning_waits_queues.msp
 - 한글 번역 완료(TechNet 웹사이트에 게시 예정)
- Ken Henderson
 - SQL Server 2005 Practical Troubleshooting
- DMVStats
 - <http://www.codeplex.com/sqldmvstats>

세미나를 마치며...

- SQL Server 2005와 관련 도구를 사용하여 진단분석 & 성능개선 프로세스를 만들자.
- 대기 및 큐 분석을 통해 시스템의 병목시점을 찾아내자.
- 모든 사건엔 증거가 있다. 그 증거는 반드시 범인을 찾아낸다.
- Focus on Execution : 알고만 있지 말고 실천하자!

- 하나님께서서는 여러분을 사랑하십니다.

저도 여러분께 도움이 되는 사람이 되도록 노력하겠습니다.
여러분과 함께 SQL로 만들어가는 좋은 세상을 이뤄가길
기대하며 노력하겠습니다.
감사합니다.

Q&A.