



데이터베이스 구성 Enterprise applications

NetApp
May 09, 2024

목차

데이터베이스 구성	1
Oracle 데이터베이스 블록 크기	1
Oracle 데이터베이스 매개 변수: db_file_multiblock_read_count	1
Oracle 데이터베이스 매개 변수: filesystemio_options	2
Oracle RAC(Real Application Clusters) 시간 초과	3

데이터베이스 구성

Oracle 데이터베이스 블록 크기

ONTAP에서는 내부적으로 가변 블록 크기를 사용하므로 원하는 블록 크기로 Oracle 데이터베이스를 구성할 수 있습니다. 그러나 파일 시스템 블록 크기가 성능에 영향을 미칠 수 있으며 경우에 따라 재실행 블록 크기가 클수록 성능이 향상될 수 있습니다.

데이터 파일 블록 크기

일부 운영 체제에서는 파일 시스템 블록 크기를 선택할 수 있습니다. Oracle 데이터 파일을 지원하는 파일 시스템의 경우 압축을 사용했을 때 블록 크기는 8KB가 됩니다. 압축이 필요하지 않은 경우 8KB 또는 4KB의 블록 크기를 사용할 수 있습니다.

데이터 파일이 512바이트 블록의 파일 시스템에 배치되면 파일 정렬 불량이 발생할 수 있습니다. LUN과 파일 시스템은 NetApp 권장 사항에 따라 적절히 정렬되었으나 파일 I/O는 정렬 불량이 될 것입니다. 이러한 정렬 불량은 심각한 성능 문제를 초래합니다.

재실행 로그를 지원하는 파일 시스템은 재실행 블록 크기의 배수인 블록 크기를 사용해야 합니다. 일반적으로 재실행 로그 파일 시스템과 재실행 로그 자체는 512바이트의 블록 크기를 사용해야 합니다.

블록 크기 재실행

재실행 속도가 빠르면 더 간소화되고 효율적인 작업으로 I/O를 수행할 수 있기 때문에 매우 빠른 재실행 속도에서 4KB 블록 크기의 성능이 향상될 수 있습니다. 재실행 속도가 50MBps보다 빠른 경우 4KB 블록 크기를 테스트하는 것을 고려해 보십시오.

고객 사례에서 4KB 블록 크기와 매우 작은 트랜잭션이 많이 수행되는 파일 시스템에서 512바이트 블록 크기의 재실행 로그를 사용하는 데이터베이스가 몇 가지 문제를 발생시키는 것으로 나타났습니다. 단일 4KB 파일 시스템 블록에 여러 512바이트 변경사항을 적용할 때 발생하는 오버헤드는 성능 문제를 야기했으며 이 문제는 512바이트의 블록 크기를 사용하도록 파일 시스템을 변경하여 해결되었습니다.



* NetApp은 관련 고객 지원 팀 또는 프로페셔널 서비스 조직에서 변경하라고 조언했거나 공식 제품 설명서에 따른 것이 아닌 한 재실행 블록 크기를 변경하지 않을 것을 권장합니다.

Oracle 데이터베이스 매개 변수: `db_file_multiblock_read_count`

를 클릭합니다 `db_file_multiblock_read_count` 매개 변수는 Oracle이 순차적 I/O 중에 단일 작업으로 읽는 Oracle 데이터베이스 블록의 최대 수를 제어합니다

하지만 이 매개 변수는 모든 읽기 작업 중에 Oracle이 읽는 블록의 수나 랜덤 I/O에 영향을 미치지 않으며 순차적 I/O의 블록 크기에만 영향을 미칩니다.

Oracle은 이 매개 변수를 설정 해제 상태로 둘 것을 권장합니다. 이렇게 하면 데이터베이스 소프트웨어가 최적의 값을 자동으로 설정할 수 있습니다. 이는 일반적으로 이 매개 변수가 1MB의 I/O 크기를 생성하는 값으로 설정된다는 뜻입니다. 예를 들어, 8KB 블록의 1MB 읽기에서는 읽을 블록이 128개 필요하며 이에 따라 이 매개 변수의 기본값은 128이 됩니다.

NetApp이 고객 사이트에서 관찰한 데이터베이스 성능 문제는 대부분 이 매개 변수의 잘못된 설정과 관련이 있습니다. Oracle 버전 8 및 9에서 이 값을 변경할 타당한 이유가 있었습니다. 그 결과, 매개 변수가 모르게 에 있을 수 있습니다 `init.ora` 데이터베이스가 Oracle 10 이상으로 업그레이드되었기 때문입니다. 128이라는 기본값에 비해 8 또는 16의 기존 설정은 순차적 I/O 성능을 크게 저하합니다.



* NetApp는 * 를 설정할 것을 권장합니다 `db_file_multiblock_read_count` 매개 변수는 에 있어서는 안 됩니다 `init.ora` 파일. NetApp이 관찰한 결과 이 매개 변수를 변경하여 성능이 개선된 경우는 없었으나 순차적 I/O 처리량 저하가 사라진 사례는 많이 있습니다.

Oracle 데이터베이스 매개 변수: `filesystemio_options`

Oracle 초기화 매개 변수 `filesystemio_options` 비동기식 및 직접 입출력의 사용을 제어합니다

일반적인 인식과 달리 비동기식 및 직접 I/O는 상호 배타적이지 않습니다. NetApp은 이 매개 변수가 고객 환경에서 종종 잘못 구성되며 이 잘못된 구성이 여러 성능 문제의 직접적인 원인이 된다는 것을 목격했습니다.

비동기식 I/O에서는 Oracle I/O 작업을 병렬화할 수 있습니다. 다양한 운영 체제에서 비동기식 I/O를 사용할 수 있게 되기 전에 사용자는 수많은 `dbwriter` 프로세스를 구성했고 서버 프로세스 구성을 변경했습니다. 비동기식 I/O를 통해 운영 체제는 매우 효율적인 병렬 방식으로 데이터베이스 소프트웨어 대신 I/O를 수행합니다. 이 프로세스는 데이터를 위험에 처하게 하지 않으며 Oracle 로깅 재실행 같은 중요한 작업은 여전히 동기식으로 수행됩니다.

직접 I/O는 운영 체제 버퍼 캐시를 우회합니다. UNIX 시스템의 I/O는 일반적으로 운영 체제 버퍼 캐시를 통해 흐릅니다. 이는 내부 캐시에서 유지되지 않는 애플리케이션에 유용하지만 Oracle은 SGA 내부에 자체 버퍼 캐시가 있습니다. 거의 모든 경우 운영 체제 버퍼 캐시를 사용하는 것보다 직접 I/O를 활성화하고 서버 RAM을 SGA에 할당하는 것이 더 낫습니다. Oracle SGA는 메모리를 더 효율적으로 사용하며 I/O가 운영 체제 버퍼를 통해 흐를 때 지연 시간을 증가시키는 추가 처리가 적용됩니다. 증가한 지연 시간은 짧은 지연 시간이 중요한 요구사항일 때 쓰기 집약적 I/O에서 특히 뚜렷하게 나타납니다.

의 옵션 `filesystemio_options` 다음과 같습니다.

- * 비동기 * Oracle은 처리를 위해 I/O 요청을 OS에 제출합니다. 이 프로세스는 Oracle I/O가 완료되기를 기다리면서 I/O 병렬화를 증가시키는 것이 아니라 Oracle이 다른 작업을 수행할 수 있도록 합니다.
- * `directio`. * Oracle은 호스트 OS 캐시를 통해 I/O를 라우팅하지 않고 물리적 파일에 직접 I/O를 수행합니다.
- * 없음. * Oracle은 동기 및 버퍼링된 I/O를 사용합니다 이 구성에서는 공유 서버와 전용 서버 프로세스 중 무엇을 선택할지 그리고 `dbwriter`의 개수가 어떻게 되는지가 더 중요합니다.
- * `SetAll`. * Oracle은 비동기 I/O와 직접 I/O를 모두 사용합니다 거의 모든 경우에 를 사용합니다 `setall` 최적의 상태.



를 클릭합니다 `filesystemio_options` 매개 변수는 DNFS 및 ASM 환경에 어떤 영향도 미치지 않습니다. DNFS 또는 ASM가 자동으로 사용되면 비동기식 및 직접 I/O 둘 다 사용됩니다

과거에는 특히 Red Hat Enterprise Linux 4(RHEL4) 릴리즈에서 비동기식 I/O 문제에 직면하는 고객이 있었습니다. 일부 오래된 인터넷 조언은 오래된 정보로 인해 비동기 IO를 피하는 것을 제안합니다. 비동기식 I/O는 현재의 모든 운영 체제에서 안정적입니다. OS에 알려진 버그가 없고 비활성화할 이유가 없습니다.

데이터베이스가 버퍼링된 I/O를 사용해온 경우 직접 I/O로 전환하면 SGA 크기 변경도 보장됩니다. 버퍼링된 I/O를 비활성화하면 호스트 운영 체제 캐시가 데이터베이스에 제공하는 성능 이점이 없어집니다. RAM을 SGA에 다시 추가하면 이 문제가 해결되며 결과적으로 I/O 성능이 개선될 것입니다.

거의 모든 경우에 Oracle SGA에 운영 체제 버퍼 캐싱보다 RAM을 사용하는 것이 더 낫지만 최고의 가치를 결정하는 것은 불가능할 수 있습니다. 예를 들어, 간헐적인 액티브 Oracle 인스턴스가 있는 데이터베이스 서버에는 아주 작은 SGA 크기의 버퍼링된 I/O를 사용하는 것이 더 나은 것입니다. 이렇게 하면 실행 중인 모든 데이터베이스 인스턴스를 통해 운영 체제의 나머지 가용 RAM을 유연하게 사용할 수 있습니다. 이는 매우 드문 상황이지만 일부 고객 사이트에서 관찰된 바 있습니다.



* NetApp는 * 설정을 권장합니다 filesystemio_options 를 선택합니다 `setall`는 어떤 상황에서는 호스트 버퍼 캐시의 손실로 인해 Oracle SGA를 늘려야 할 수 있음을 유념하십시오.

Oracle RAC(Real Application Clusters) 시간 초과

Oracle RAC는 클러스터 상태를 모니터링하는 여러 유형의 내부 하트비트 프로세스를 갖춘 클러스터웨어 제품입니다.



의 정보 "**잘못 장착되었습니다**" 섹션에는 네트워크 스토리지를 사용하는 Oracle RAC 환경에 대한 중요 정보가 포함되어 있으며, 대부분의 경우 RAC 클러스터가 네트워크 경로 변경 및 스토리지 페일오버 /스위치오버 작업을 지속할 수 있도록 기본 Oracle RAC 설정을 변경해야 합니다.

디스크 시간 초과

은 운영 스토리지와 관련된 주요 RAC 매개 변수입니다 `disktimeout`. 이 매개 변수는 voting 파일 I/O가 완료되어야 하는 임계값을 제어합니다. 를 누릅니다 `disktimeout` 매개 변수가 초과되면 클러스터에서 RAC 노드가 제거됩니다. 이 매개 변수의 기본값은 200이며 이 값은 표준 스토리지 테이크오버 및 반환 절차를 진행하는 데 충분합니다.

많은 요소가 테이크오버나 반환에 영향을 미치기 때문에 RAC 구성을 운영 환경에 배치하기 전에 철저히 테스트하는 것이 좋습니다 NetApp. 스토리지 페일오버가 완료되는 데 필요한 시간 외에 링크 통합 제어 프로토콜(LACP) 변경사항을 전달하기 위한 추가 시간도 필요합니다. 또한 SAN 다중 경로 소프트웨어는 I/O 시간 초과를 감지하고 대체 경로를 다시 시도해야 합니다. 데이터베이스가 매우 활성화된 경우 보팅 디스크 I/O가 처리되기 전에 많은 양의 I/O를 대기열에 넣고 다시 시도해야 합니다.

실제 스토리지 테이크오버 또는 반환을 수행할 수 없을 경우 데이터베이스 서버에서 케이블을 빼는 테스트를 통해 그 영향을 시뮬레이션할 수 있습니다.



- NetApp는 * 다음을 권장합니다.
- 를 종료합니다 `disktimeout` 기본값 200의 매개 변수입니다.
- 항상 RAC 구성을 철저히 테스트하십시오.

잘못 장착되었습니다

를 클릭합니다 `misscount` 매개 변수는 일반적으로 RAC 노드 간 네트워크 하트비트에만 영향을 미치며 기본값은 30초입니다. 그리드 바이너리가 스토리지 어레이에 있거나 운영 체제 부트 드라이브가 로컬에 있지 않은 경우 이 매개 변수의 중요성이 커집니다. 부트 드라이브가 FC SAN에 있는 호스트, NFS 부팅 운영 체제 그리고 VMDK 파일처럼 부트 드라이브가 가상화 데이터 저장소에 있는 호스트가 여기에 포함됩니다.

스토리지 테이크오버 또는 반환에 의해 부트 드라이브에 대한 액세스가 중단되면 그리드 바이너리 위치 또는 전체 운영 체제가 일시적으로 멈출 수 있습니다. ONTAP에서 스토리지 작업이 완료되고 운영 체제에서 경로가 변경되고 I/O가 재개되는 데 필요한 시간이 을 초과할 수 있습니다 `misscount` 임계값. 노드는 부트 LUN 또는 그리드 바이너리에 대한 연결이 복원되는 즉시 제거됩니다. 대부분의 경우 재부팅 이유를 나타내는 로깅 메시지 없이 제거 및 후속 재부팅 동작이

수행합니다. 모든 구성이 영향을 받는 것은 아니므로 부트 드라이브에 대한 통신이 중단되는 경우 RAC가 안정된 상태를 유지하도록 RAC 환경에서 SAN 부팅, NFS 부팅 또는 데이터 저장소 기반 호스트를 테스트하십시오.

비로컬 부팅 드라이브 또는 비로컬 파일 시스템 호스팅의 경우 grid 바이너리 misscount 일치시키려면 변경해야 합니다 disktimeout. 이 매개 변수가 변경되면 추가 테스트를 수행하여 노드 페일오버 시간 등 RAC 동작에 미치는 영향도 파악하십시오.



- NetApp는 * 다음을 권장합니다.
- 를 그대로 둡니다 misscount 다음 조건 중 하나가 적용되지 않는 한 매개 변수 기본값은 30입니다.
 - grid 바이너리는 NFS, iSCSI, FC 및 데이터 저장소 기반 드라이브를 포함하여 네트워크 연결 드라이브에 있습니다.
 - 운영 체제가 SAN 환경에서 부팅됩니다.
- 이러한 경우 OS 또는 에 대한 액세스에 영향을 미치는 네트워크 중단의 영향을 평가합니다 GRID_HOME 파일 시스템 경우에 따라 이러한 중단으로 인해 Oracle RAC 데몬에 지연이 발생하며 이로 인해 로 이어질 수 있습니다 misscount `시간 초과 및 제거 기반. 시간 초과 기본값은 27초이며 이 값은 입니다 `misscount 빼기 reboottime. 이 경우 을 증가시킵니다 misscount 200을 선택하십시오 disktimeout.

저작권 정보

Copyright © 2024 NetApp, Inc. All Rights Reserved. 미국에서 인쇄된 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그래픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이선스와 고지사항이 적용됩니다.

본 소프트웨어는 NetApp에 의해 '있는 그대로' 제공되며 상품성 및 특정 목적에의 적합성에 대한 명시적 또는 묵시적 보증을 포함하여(이에 제한되지 않음) 어떠한 보증도 하지 않습니다. NetApp은 대체품 또는 대체 서비스의 조달, 사용 불능, 데이터 손실, 이익 손실, 영업 중단을 포함하여(이에 국한되지 않음), 이 소프트웨어의 사용으로 인해 발생하는 모든 직접 및 간접 손해, 우발적 손해, 특별 손해, 징벌적 손해, 결과적 손해의 발생에 대하여 그 발생 이유, 책임론, 계약 여부, 엄격한 책임, 불법 행위(과실 또는 그렇지 않은 경우)와 관계없이 어떠한 책임도 지지 않으며, 이와 같은 손실의 발생 가능성이 통지되었다 하더라도 마찬가지입니다.

NetApp은 본 문서에 설명된 제품을 언제든지 예고 없이 변경할 권리를 보유합니다. NetApp은 NetApp의 명시적인 서면 동의를 받은 경우를 제외하고 본 문서에 설명된 제품을 사용하여 발생하는 어떠한 문제에도 책임을 지지 않습니다. 본 제품의 사용 또는 구매의 경우 NetApp에서는 어떠한 특허권, 상표권 또는 기타 지적 재산권이 적용되는 라이선스도 제공하지 않습니다.

본 설명서에 설명된 제품은 하나 이상의 미국 특허, 해외 특허 또는 출원 중인 특허로 보호됩니다.

제한적 권리 표시: 정부에 의한 사용, 복제 또는 공개에는 DFARS 252.227-7013(2014년 2월) 및 FAR 52.227-19(2007년 12월)의 기술 데이터-비상업적 품목에 대한 권리(Rights in Technical Data -Noncommercial Items) 조항의 하위 조항 (b)(3)에 설명된 제한사항이 적용됩니다.

여기에 포함된 데이터는 상업용 제품 및/또는 상업용 서비스(FAR 2.101에 정의)에 해당하며 NetApp, Inc.의 독점 자산입니다. 본 계약에 따라 제공되는 모든 NetApp 기술 데이터 및 컴퓨터 소프트웨어는 본질적으로 상업용이며 개인 비용만으로 개발되었습니다. 미국 정부는 데이터가 제공된 미국 계약과 관련하여 해당 계약을 지원하는 데에만 데이터에 대한 전 세계적으로 비독점적이고 양도할 수 없으며 재사용이 불가능하며 취소 불가능한 라이선스를 제한적으로 가집니다. 여기에 제공된 경우를 제외하고 NetApp, Inc.의 사전 서면 승인 없이는 이 데이터를 사용, 공개, 재생산, 수정, 수행 또는 표시할 수 없습니다. 미국 국방부에 대한 정부 라이선스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

NETAPP, NETAPP 로고 및 <http://www.netapp.com/TM>에 나열된 마크는 NetApp, Inc.의 상표입니다. 기타 회사 및 제품 이름은 해당 소유자의 상표일 수 있습니다.