

NBD Bench 시험 결과 보고서

(NBD Bench Test Result)

문서 정보

본 보고서는 CUBRID 2008 R1.1 버전의 성능 측정을 위해 NBD Bench 시험을 수행하고 수행 결과를 정리한 결과 보고서이다. 객관적인 성능 비교를 위해 일반적으로 많이 사용되고 있는 세 개의 DBMS를 선정하여 CUBRID의 시험 결과와 비교하였다.

문서 작성자

관련 버전	일자	이력사항	작성자
2008R1.1	2008.11.20	처음 작성	최웅세(wschoi@nhncorp.com)

저작권

Copyright 2008 Search Solution Corporation. All Rights Reserved.

이 문서는 서치솔루션㈜의 지적 자산이며, 정보제공의 목적으로만 제공됩니다.

서치솔루션㈜는 이 문서에 수록된 정보의 완전성과 정확성을 검증하기 위해 노력하였으나, 발생할 수 있는 내용상의 오류나 누락에 대해서는 책임지지 않습니다. 따라서 이 문서를 수정하고 배포하는 것은 자유로우나,

이 문서의 사용이나 사용 결과에 따른 책임은 전적으로 사용자에게 있으며, 서치솔루션㈜는 이에 대해 명시적 혹은 묵시적으로 어떠한 보증도 하지 않습니다.

관련 URL 정보를 포함하여 이 문서에서 언급한 특정 소프트웨어 상품이나 제품은 해당 소유자의 저작권법을 따르며, 해당 저작권법을 준수하는 것은 사용자의 책임입니다.

서치솔루션㈜는 이 문서의 내용을 예고 없이 변경할 수 있습니다.

목차

1. 목적	4
2. 시험 시나리오	6
2.1 대상 데이터베이스 구성	7
2.2 스키마	7
2.3 워크로드	10
2.4 시험 방식	13
3. 시험 환경	15
3.1 시험 환경	16
4. 시험 결과 분석 및 결론	17
4.1 DBMS 별 성능 분석	18
4.2 데이터베이스 규모와 성능과의 관계	18
5. 결론	20
첨부 1: NBD 벤치마크 규격 요약	22
첨부 2: 벤치마크 시험 결과	27

1. 목적

본 시험에서는 CUBRID 2008 R1.1 DBMS 의 성능 검증을 위해 NBD 벤치마크(NHN Internet Bulletin Board Application Database Benchmark)를 수행하여 CUBRID 의 서비스 성능을 OSS DBMS D1, 상용 DBMS D1, 상용 DBMS D2 와 비교한다.

NBD 벤치마크에 대해서는 NDB 규격 문서를 참조하기 바라며, 간략한 요약 정보는 첨부 1: NBD 벤치마크 규격 요약을 참조하기 바랍니다.

2. 시험 시나리오

2.1 대상 데이터베이스 구성

시험에 사용될 데이터베이스 구성은 소규모, 중규모 등 2 가지 유형이며, 아래 <표 1> 대상 데이터베이스 규모에 각 데이터베이스의 특징이 기술되어 있다.

<표 1> 대상 데이터베이스 구성

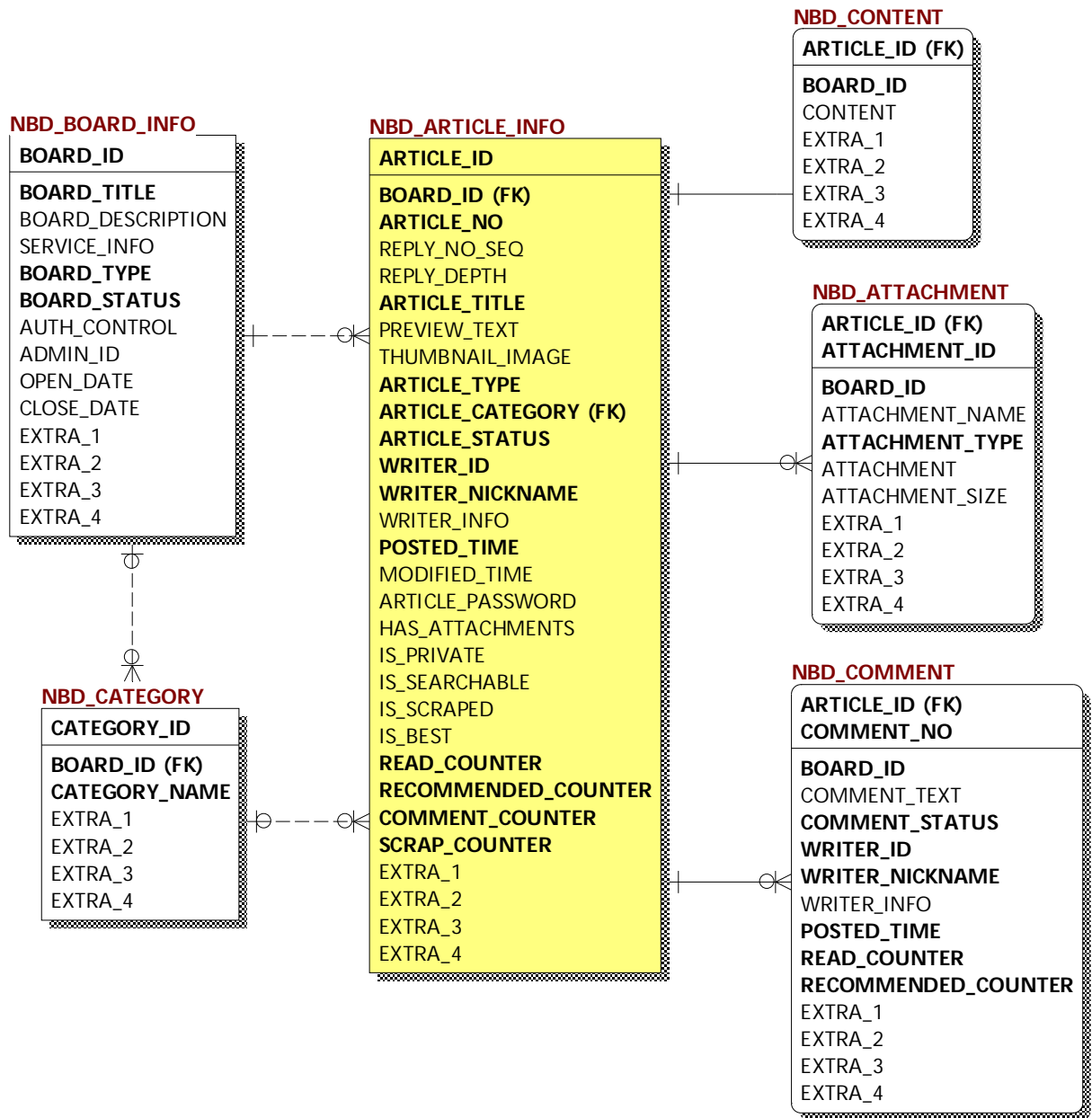
규모	게시판 수		총 row 수 (오차 범위 ± 10%)	응용
	소	중		
소규모(레벨 1)	1	1	101,000	공지사항
중규모(레벨 2)	2	6	602,000	UCC 게시판 중 중소형 규모

2.2 스키마

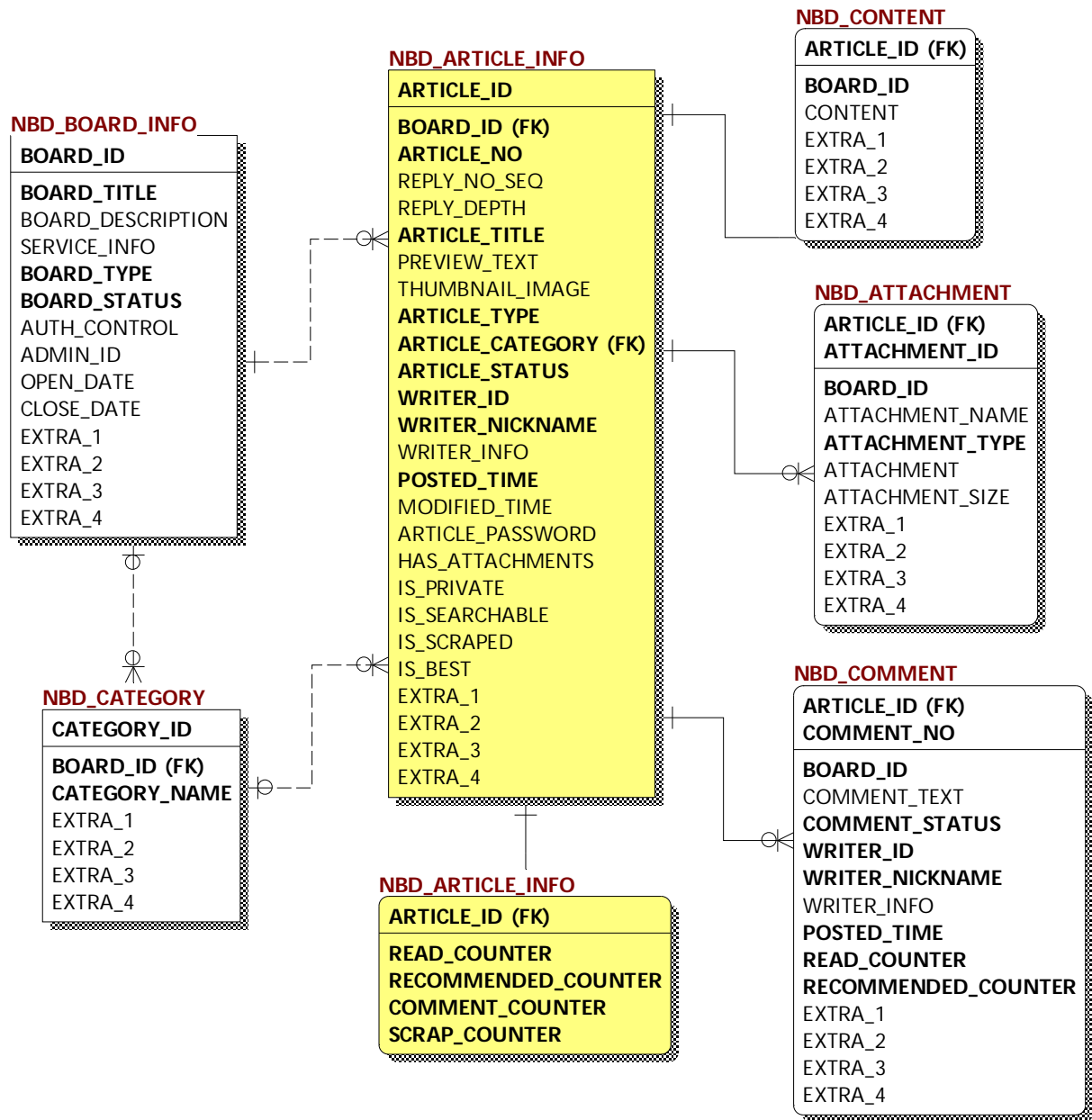
시험에 사용된 스키마는 첨부 1: BBD 벤치마크 규격 요약에 기록된 것과 동일하나, OSS DBMS D1 의 성능 튜닝을 위해 게시물의 조회수를 세는 필드를 분리한 경우를 별도로 시험한다. OSS DBMS D1 의 경우 Query Result Cache 기능을 이용한 경우와 그렇지 않은 경우 성능 차이가 아주 크다.

그러나, 현재 NBD 규격과 같이 게시물정보 테이블에 조회수를 같이 관리하게 되는 경우 매번 조회수가 변경됨으로 인해 OSS DBMS D1 의 Query Result Cache 기능이 제 역할을 못하고 매번 다시 결과 집합을 처리해야 한다. 따라서 본 시험은 조회수 필드를 별도의 테이블로 분리한 스키마와, NBD 규격과 동일하게 게시물 정보와 통합한 스키마에 대해 각각 비교 시험을 진행한다.

아래 <그림 1> 스키마 통합 모델은 NBD 규격에서 정의한 스키마 모델이다. 즉, 조회수 필드가 게시물 정보 테이블(NBD_ARTICLE_INFO)의 속성으로 관리되는 모델이다. <그림 2> 스키마 분리 모델은 OSS DBMS D1 의 성능 개선을 위해 조회수를 별도의 테이블로 분리하여 게시물 정보 테이블(NBD_ARTICLE_INFO)에만 접근하는 질의에 대해서는 Query Result Cache 를 사용할 수 있도록 한 것이다.



<그림 1> 스키마 통합



<그림 2> 스키마 분리 모델

각 테이블 별로 설정된 인덱스 정보는 <표 2> 인덱스 구성과 같다. 인덱스는 각 DBMS 시스템 별로 최적의 성능을 낼 수 있도록 구성되어 있으며, 상용 DBMS D1, 상용 DBMS D2, OSS DBMS D1의 경우에는 CHINA 데이터 분석팀의 DBA들이 튜닝 작업을 진행하면서 설정하였다.

<표 2> 인덱스 구성

테이블 명	CUBRID, 상용 DBMS D1, D2 DB		OSS DBMS D1	
	인덱스유형	대상 칼럼	인덱스유형	대상 칼럼
NBD_BOARD_INFO	PK	BOARD_ID	PK	BOARD_ID

NBD_CATEGORY	PK	CATEGORY_ID	PK	CATEGORY_ID
NBD_ARTICLE_INFO	PK	ARTICLE_ID	PK	ARTICLE_ID
	UNIQUE	BOARDID ARTICLE_CATEGORY ARTICLE_NO REPLY_DEPTH REPLY_NO_SEQ POSTED_TIME	UNIQUE	BOARDID ARTICLE_CATEGORY ARTICLE_NO REPLY_DEPTH REPLY_NO_SEQ POSTED_TIME
	일반	BOARD_ID ARTICLE_CATEGORY	일반	BOARD_ID ARTICLE_CATEGORY
	일반	BOARD_ID ARTICLE_CATEGORY ARTICLE_NO	일반	BOARD_ID ARTICLE_CATEGORY ARTICLE_STATUSIS_PRIVATE ARTICLE_NO REPLY_DEPTH REPLY_NO_SEQ POSTED_TIME
	REVERSE	BOARD_ID ARTICLE_CATEGORY ARTICLE_NO		
NBD_CONTENT	PK	ARTICLE_ID	PK	ARTICLE_ID
NBD_ATTACHMENT	PK	ARTICLE_ID, ATTACHMENT_NO	PK	ARTICLE_ID, ATTACHMENT_NO
NBD_COMMENT	PK	ARTICLE_ID, COMMENT_NO	PK	ARTICLE_ID, COMMENT_NO
	일반	ARTICLE_ID POSTED_TIME	일반	ARTICLE_ID COMMENT_STATUS POSTED_TIME

2.3 워크로드

NBD 벤치에 정의된 두 가지 워크로드 중 HOTSPOT READ 유형을 시험한다. HOTSPOT READ 유형은 특정 게시물에 부하가 집중되는 경우를 모델링한 것이다. 포탈의 메인 페이지에 게시되는 글의 경우에는 해당 글에 집중적으로 사용자 클릭이 발생하면서, DBMS 단에서는 해당 게시물을 읽는 작업과 게시물의 조회수를 증가시키는 작업이 동시에 이루어져야 한다. 이 때 게시물의 조회수를 증가시키는 갱신 작업이 DBMS에 많은 부하를 일으키며 성능의 병목이 된다. 본 시험에서는 이러한 경우를 시뮬레이션 하여 CUBRID와 상용 DBMS D1, 상용 DBMS D2, OSS DBMS D1의 시스템 측면의 특성 및 성능 분석을 진행하고자 한다.

다음 <표 3> 워크로드 요약은 워크로드의 트랜잭션을 요약한 것이다.

<표 3> 워크로드 요약

트랜잭션 믹스	요약	단위 작업	대기 시간	비율
MIX1	게시물 읽기	게시물 정보 조회 조회수 증가 콘텐츠 조회 첨부 파일 출력 이전 글 다음 글 출력 답 글 출력	5ms	70%
MIX2	게시물 읽기 + 목록 출력	게시물 정보 조회 조회수 증가 콘텐츠 조회 첨부 파일 출력 이전 글 다음 글 출력 답 글 출력 게시물 포함 목록 출력	5ms	20%
MIX3	게시물 목록 조회	목록 출력	50ms	10%

대상 HOTSPOT 게시물은 6 개 (현재 네이버 서비스에서 한 서비스 당 메인 페이지에 노출되는 게시물은 평균 5~6 개 정도임)

<표 4> CUBRID 파라미터 설정 값

항목	값	비고
버전	CUBRID 2008 R1.1	
Connections	max_clients 200	
Isolation Level	isolation_level=TRAN_REP_CLASS_UNCOMMIT_INSTANCE	Dirty read 허용
Memory Buffer	data_buffer_pages=2000	
Lock Timeout	lock_timeout_in_secs=-1	
Logging	log_buffer_pages=100 async_commit=on group_commit_interval_in_msecs=5 checkpoint_evey_npages=10000 checkpoint_interval_in_mins=30	비동기식 Commit, Group Commit, 로그 diff & compression
Thread		
Query processing	sort_buffer_pages=32 temp_file_memory_size_in_pages=4 index_scan_in_oid_order=off max_plan_cache_entries=200 query_cache_mode=2	Query Plan Cache, Query Result Cache
Broker	SQL_LOG=off JDBC_CACHE=ON	

<표 5> 상용 DBMS D1 파라미터 설정 값

항목	값	비고
버전	-	
Connections	processes =500	
Isolation Level	transaction_isolation= READ-COMMITTED	Default isolation level, can't change it
Memory Buffer	10g: db_buffer : 2.5G sga_target: 4G 11g: db_buffer : 2.5G memory_target : 4G	
Lock Timeout	default	
Logging	logging	
Result_cache	result_cache_mode=manual	
Other parameter	Commit_logging=batch commit_write=nowait	

<표 6> OSS DBMS D1 파라미터 설정 값

항목	설정 값	비고
버전	-	
Connections	max_connections 500	
Isolation Level	transaction_isolation= READ-UNCOMMITTED	Dirty Read 허용
Memory Buffer	innodb_buffer_pool_size = 4G innodb_additional_mem_pool_size = 16M	실제 가용 메모리의 50% 정도
Lock Timeout	innodb_lock_wait_timeout = 60	
Logging	innodb_log_buffer_size = 8M innodb_log_file_size = 512M innodb_log_files_in_group = 3	로그 파일을 3개로 나누어 기록
Thread	thread_cache_size=128 innodb_thread_concurrency = 4	
Query processing	table_cache = 1024 sort_buffer_size = 16M join_buffer_size = 16M read_buffer_size = 16M read_rnd_buffer_size = 16M query_cache_size = 64M query_cache_limit = 2M	Query Result Cache를 사 용하며, sort나 join이 disk operation을 일으키지 않 도록 충분한 크기 설정

<표 7> 상용 DBMS D2 파라미터 설정 값

항목	설정 값	비고
버전	-	
Connections	max_connections = 0	제한 없음
Isolation Level	transaction_isolation= READ-UNCOMMITTED	저장 프로시저에서 명시
Memory Buffer	min server memory = 1638M	

	max server memory = 3072M	
Lock Timeout	query wait = -1	계속 기다림
Logging		
Thread	max worker threads = 255	
Query processing		

2.4 시험 방식

SQLMAP 인터페이스를 통해 10 분 동안 지속적으로 부하를 발생시키는 시험을 3 회 연속 시행 후 중간 값을 선택한다. NBD 규격의 4.4 절에 정의된 Measuring Rules 에 따라 다음과 같은 조건을 만족시켜야 한다.

- Steady State Period 동안 성능 값을 측정하며, Expected Response Time 은 0.3 초 이내여야 한다.
- 테스트 시행 중 DB 서버의 CPU 사용량은 85% 이내여야 한다.
- 각 DBMS 시스템이 최적의 성능의 낼 수 있도록 부하량을 조절할 수 있다.
- 트랜잭션 믹스에서 하나의 트랜잭션이 실패하면, 다음 순서의 트랜잭션들은 수행되지 않으며, 해당 서비스는 실패로 처리된다.
- 트랜잭션의 실패율은 10% 이내여야 한다.
- 측정값은 최소한 매 10 초 단위로 수집되어야 한다.
- Warm-up time 은 측정 시간에서 제외 가능하다.

SQLMAP 인터페이스 설정은 모든 DBMS 에 공통적으로 다음 설정과 동일하다

```
<settings
  cacheModelsEnabled="true"
  enhancementEnabled="true"
  lazyLoadingEnabled="true"
  maxRequests="512"
  maxSessions="128"
  maxTransactions="128"
  useStatementNamespaces="false"
  defaultStatementTimeout="0" />
  <transactionManager type="JDBC">
    <dataSource type="DBCP">
      <property name="maxActive" value="100" />
      <property name="minIdle" value="10" />
```

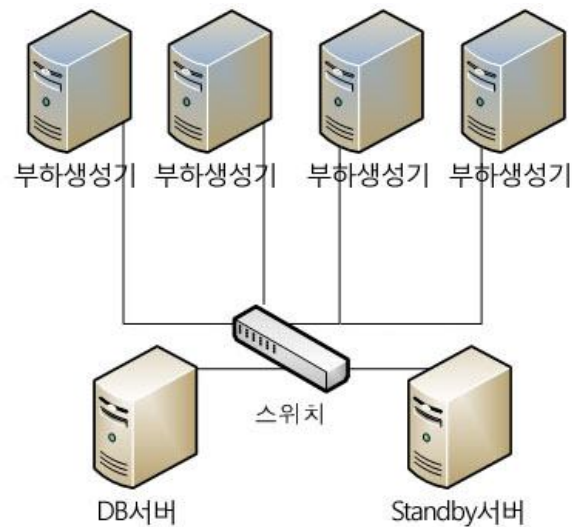
```
<property name="maxIdle" value="-1" />  
<property name="maxWait" value="60000" />  
<property name="poolPreparedStatements" value="true" />  
</dataSource>  
</transactionManager>
```

3. 시험 환경

본 장에서는 시험이 진행될 테스트베드의 구성에 대해 간략하게 기술한다. 테스트베드의 각 장비 별 규격 및 특징은 2.5 절의 **오류! 참조 원본을 찾을 수 없습니다.**를 참고하기 바란다.

3.1 시험 환경

시험 환경은 HA 구성을 하지 않고 DB 서버만 구성한 후 시험하였다. 또한 DB 서버에 최대한 부하를 주기 위하여 부하 서버는 4 대로 구성하였고 각각의 부하 서버는 동시 사용자 40 명의 부하를 주어 DB 서버에 가해지는 전체 부하는 동시 160 명을 모델링 하였다. 즉, 각 부하 서버 별로 부하 생성 프로그램 2 개를 구동하고, 각 프로그램은 다시 20 개의 쓰레드를 생성하여 DBMS 에 부하를 준다. CUBRID 의 경우 DB 서버와 브로커를 분리할 수 있으므로 복제 상태에서의 stand-by 서버에 브로커를 분리한 경우로 가정하여 수행하였다.

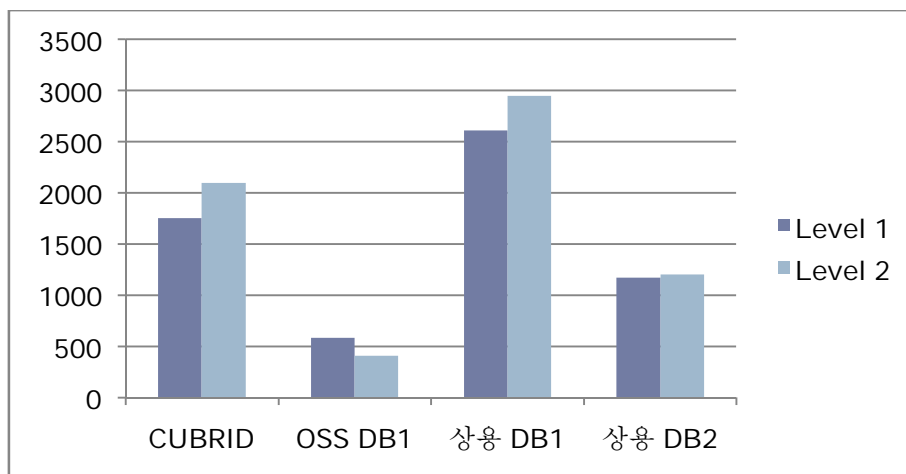


<그림 3> 시험 환경

4. 시험 결과 분석 및 결론

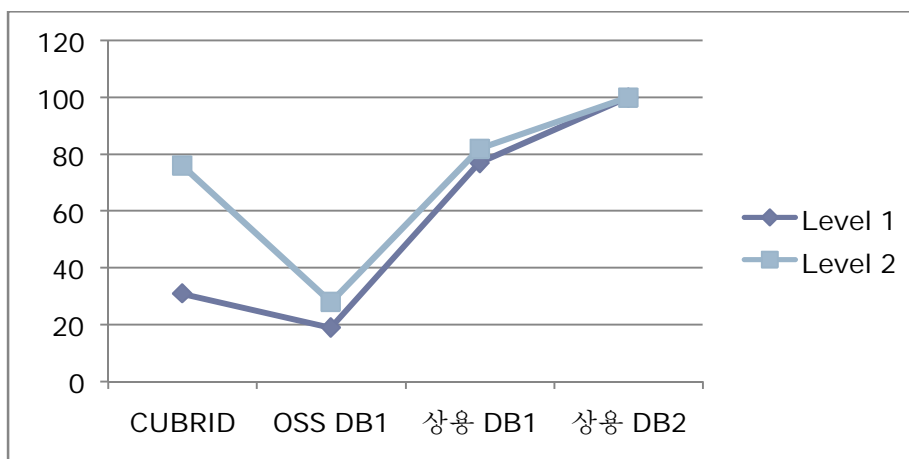
본 장에서는 진행된 시험 결과를 분석하고 그에 대한 결론을 내린다. 시험 결과에 대한 자세한 수치 정보는 첨부2: 벤치마크 시험 결과를 참고하길 바란다.

4.1 DBMS 별 성능 분석



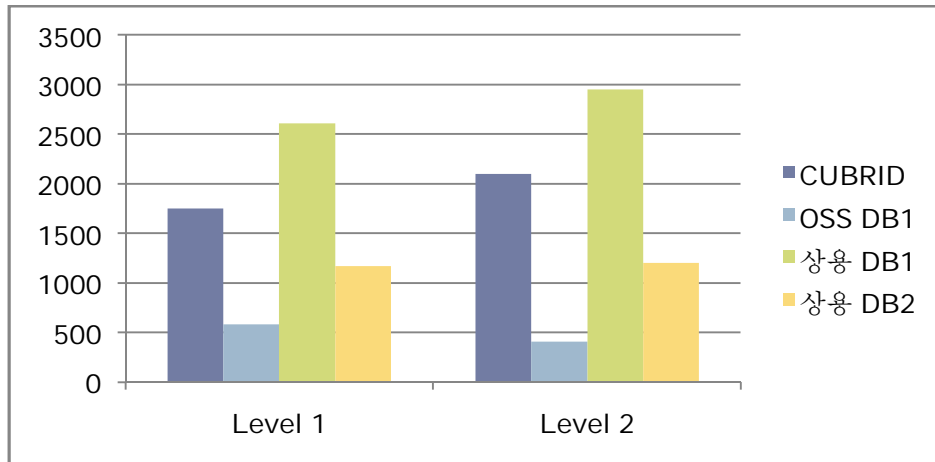
<그림 4> DBMS 별 성능 변화

<그림 4>는 동시 사용자 160 에서 DBMS 별 성능 수치를 표시한 그래프이다. 각 DBMS 별 성능 수치는 <그림 5>의 CPU 리소스 사용량과 거의 일치하는 모습을 보이고 있다.



<그림 5> DBMS 별 CPU 사용률 변화

4.2 데이터베이스 규모와 성능과의 관계



<그림 6> 데이터베이스 규모에 따른 성능 변화

5. 결론

이번 성능 시험은 데이터베이스 규모, 부하의 증가 및 각 데이터베이스가 제공하는 cache 기능의 여러 변수를 이용하여 시행되었고 각각의 변수에 따라 나타나는 네 가지 DBMS의 변화를 관찰하고 그에 따른 분석을 통하여 네 가지 DBMS의 장단점을 살펴보았다.

상용 DBMS D1의 경우 소규모, 중규모 벤치마킹에서 가장 뛰어난 성능 수치를 보여주고 있다.

CUBRID의 경우 소규모 및 중규모에서 상용 DBMS D1에 이어 두 번째 성능을 보이고 있다. 여기서 한 가지 의문 사항은 소규모의 경우 CPU 사용률이 30% 이내로 부하를 더 가할 경우 더 높은 성능을 보여줄 가능성이 있다. 그러나 중규모에서 CPU 사용률이 80% 정도의 높은 이용률을 보인 점은 좀 더 분석이 필요하다.

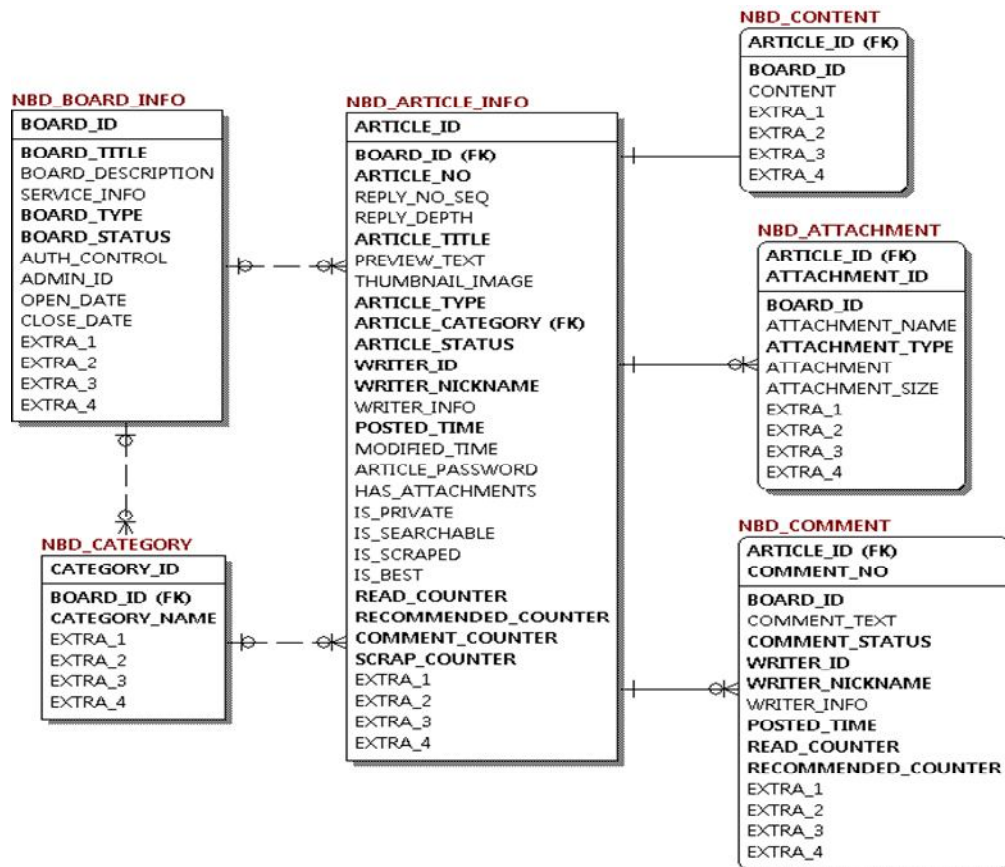
D2 DB의 경우 소규모, 중규모 모두 CPU 사용률이 모두 100%로 최대 성능 수치를 보인 것으로 판단된다.

OSS DBMS D1 서버의 경우 소규모, 중규모 모두 30% 이내의 낮은 사용률을 보이고 있다. 부하가 작은 경우에도 비슷한 이용률을 보이고 있는 점은 추가 분석을 통해 분석해야 할 부분이다.

첨부 1: NBD 벤치마크 규격 요약

NBD 벤치마크는 게시판 류의 웹 서비스에 사용된 DBMS의 성능을 서비스 관점에서 측정하기 위한 벤치마크로서 하드웨어와 DBMS의 설정에는 제약을 두지 않으며, Pricing을 산출하지 않으므로 동일 하드웨어 구성에서 DBMS 간 성능에 대한 상호 비교가 필요한 경우 유용하게 사용될 수 있다.

NBD 벤치마크 시험을 위해 구축되어야 하는 데이터베이스의 논리 스키마는 <그림 7> 과 같다.



<그림 7> NBD의 논리 E-R 다이어그램

다음 <표 8>은 NBD 데이터베이스의 구성 단위를 게시판 단위로 정의한 것이다.

<표 8> NBD 구성 단위 및 확장 규칙

게시판유형	소규모	중규모	대규모	비고
게시물 수	1,000	100,000	10,000,000	± 10%
카테고리	없음	10개 미만	10개 미만	
게시물 유형	TEXT	TEXT, IMAGE, 동영상 등	TEXT, IMAGE, 동영상 등	
게시물 크기	90%이상 : 1K	50% : 1K 이하 30% : 1K ~ 10K 20% : 10K ~ 64K	50% : 1K 이하 30% : 1K ~ 10K 20% : 10K ~ 64K	
게시물 작성자 수	1명	1명 당 평균 10개 글 작성	1명 당 평균 10개 글 작성	
첨부 파일 수	없음	50% : 5개	50% : 5개	
덧글 수	없음	50% : 10개 40% : 없음 10% : 100개	50% : 10개 40% : 없음 10% : 100개	
답글	없음	전체 게시물 중 10%가 답글	전체 게시물 중 10%가 답글	
비고	공지사항 류			

NBD 데이터베이스는 <표 8>의 각 단위 구성 요소를 조합하여 계속 확장 할 수 있다. 확장 규칙은 다음과 같다.

- 단계 1 은 소규모 게시판 1 개로 이루어진다.
- 단계 5 까지는 소규모 게시판 1 개에 중간규모 게시판을 5 개 단위로 늘린다.
- 단계 6 부터는 소규모 게시판 1 개, 중간규모 게시판 5 개에 대규모 게시판을 2 개 단위로 늘린다.

NDB 워크로드는 크게 유형 1 과 유형 2 가 정의되어 있다. 유형 1 은 HOTSPOT READ 모델로서, 특정 게시물에 조회가 집중되는 경우를 모델링한 것이다. 인터넷 서비스의 경우 메인 페이지에 게시물이 등록되면 많은 사용자가 몇 개의 페이지를 집중적으로 조회하는 경우나, 인기 글 목록을 별도로 분리하여 게시하는 경우가 유형 1 과 유사한 경우이다. 유형 2 는 NAVIGATION 모델로서, 게시물에 대한 접근이 분산되는 경우이다. 카페, 블로그와 같은 서비스가 유형 2 에 속한다.

다음 <표 9>는 워크로드 유형 별 특징을 요약한 것이다.

<표 9> 워크로드 유형 별 특징

구분	워크로드 유형1 (Hotspot Read)	워크로드 유형2 (Navigation)
개요	지정된 수의 특정 게시 글을 주어진 시간 동안 '게시물읽기'-'목록보기'를 연속 수행 (특정 인기 게시물에 부하가 집중되는 경우를 모델링)	'목록보기'를 주로 수행하면서 '게시물읽기'와 '추천하기'가 계속되는 경우를 모델링 (주기적으로 인기 글 선정과 선택이 수행됨)
트랜잭션 믹스	게시판ID와 게시물ID로 조회수 갱신, 글 내용조회, 첨부목록조회, 덧글 목록 조회, 이전 글 다음 글 조회, 답글 목록 조회 수행 (대기시간 5msec)	게시판ID로 등록시간 순 목록조회-1페이지, 등록 시간 순 목록조회-2페이지, 등록시간 순 목록조회-3페이지, 조회수역순 목록조회-1페이지, 추천수 역순

	게시판ID와 게시물ID로 조회수 갱신, 글 내용조회, 첨부목록조회, 댓글 목록 조회, 이전 글 다음 글 조회, 답글 목록 조회, 등록 순 목록 조회 수행 (대기시간 5msec) 게시판ID와 페이지 번호 1로 등록시간 순 목록 조회 수행 (대기 시간 50 msec)	목록조회-1페이지 수행 (작업 간 대기시간 10msec) 게시판ID로 등록 시간 순 목록조회-1페이지, 게시물 읽기 등 NBD 3.8.1에 명시된 작업 수행 게시판ID로 게시물 갱신, 게시물 조회 수행 (대기시간 30초)
파라미터	전체 수행 시간: 10분/30분/60분 데이터베이스 규모: 최소 1단계 구성 이상 트랜잭션 믹스 별 수행 수: 워크로드 전체에서 사용되는 Hotspot 게시물 수: 3,6,9,12	전체 수행 시간: 10분/30분/60분 데이터베이스 규모: 최소 2단계 구성 이상 트랜잭션 믹스 별 수행 수: 최소 10개 이상
요구사항	트랜잭션 믹스 1, 2, 3을 7:2:1의 비율로 정해진 시간 동안 반복 실행 Hotspot 게시물들은 서로 인접한 게시물이 아니며, 첫 번째 및 마지막 게시물이 아님. (인접 글 있음) 모든 Hotspot 게시물은 3개 이상의 첨부와 50개 이상의 댓글, 그리고 1개 이상의 답글을 가지고 있어야 함 Hotspot 게시물들은 트랜잭션 믹스 1, 2에 순차적으로 배분 트랜잭션 믹스 3은 Hotspot 게시물이 속한 게시판 중에 하나를 선택하여 수행 (페이지 당 게시물 목록 수는 20)	트랜잭션 믹스 1과 2는 동일 비율로 정해진 시간 동안 반복, 트랜잭션 믹스 3은 1개만 정해진 시간 동안 반복 실행 트랜잭션 믹스 1과 2에서 페이지 당 게시물 목록 수는 20 트랜잭션 믹스 1은 최소한 3개 페이지 이상으로 구성되는 게시판을 중복 없이 사용 트랜잭션 믹스 2는 중간규모 게시판들에서 정규 분포에 따라 선택하여 실행 트랜잭션 믹스 2에서 게시물과 댓글의 선택은 고정 되면 안됨 트랜잭션 믹스 3은 모든 게시판에 대해 순차적으로 실행

NBD 벤치마크에서 워크로드 수행 시 일반적 제약 사항은 다음과 같다.

- 모든 트랜잭션의 ACID(Atomicity, Consistency, Isolation, Durability)가 보장되어야 한다.
- 게시물을 읽을 때 증가하는 조회수는 곧바로 DB 에 반영되어야 하며, 다른 트랜잭션에서 조회했을 때 항상 정확한 값을 제공해야 한다.
- Lock Time Out 및 Deadlock 은 발생 가능하나 전체 트랜잭션의 10% 미만이어야 한다.

NBD 벤치마크 시험 결과에는 다음 표 10 과 같은 항목에 대한 측정값이 포함되어야 하며 다음과 같은 요구 사항을 만족 시켜야 한다.

- 측정값은 Steady State Period 동안에 측정되어야 한다.
- 트랜잭션 믹스에서 하나의 트랜잭션이 실패하면, 다음 순서의 트랜잭션들은 수행되지 않으며, 해당 서비스는 실패로 처리된다.
- 트랜잭션의 실패율은 10% 이내여야 하고 최고 CPU 사용량은 85% 이내여야 한다.
- 측정값은 최소한 매 10 초 단위로 수집되어야 한다.
- Warm-up time 은 측정 시간에서 제외 가능하다.
- Expected response time 은 0.3 초 이내이다.

- 최종 성능은 시스템 별로 처리한 초당 페이지 수(PV/sec)로 측정한다.

<표 10> NBD 결과 측정 방법

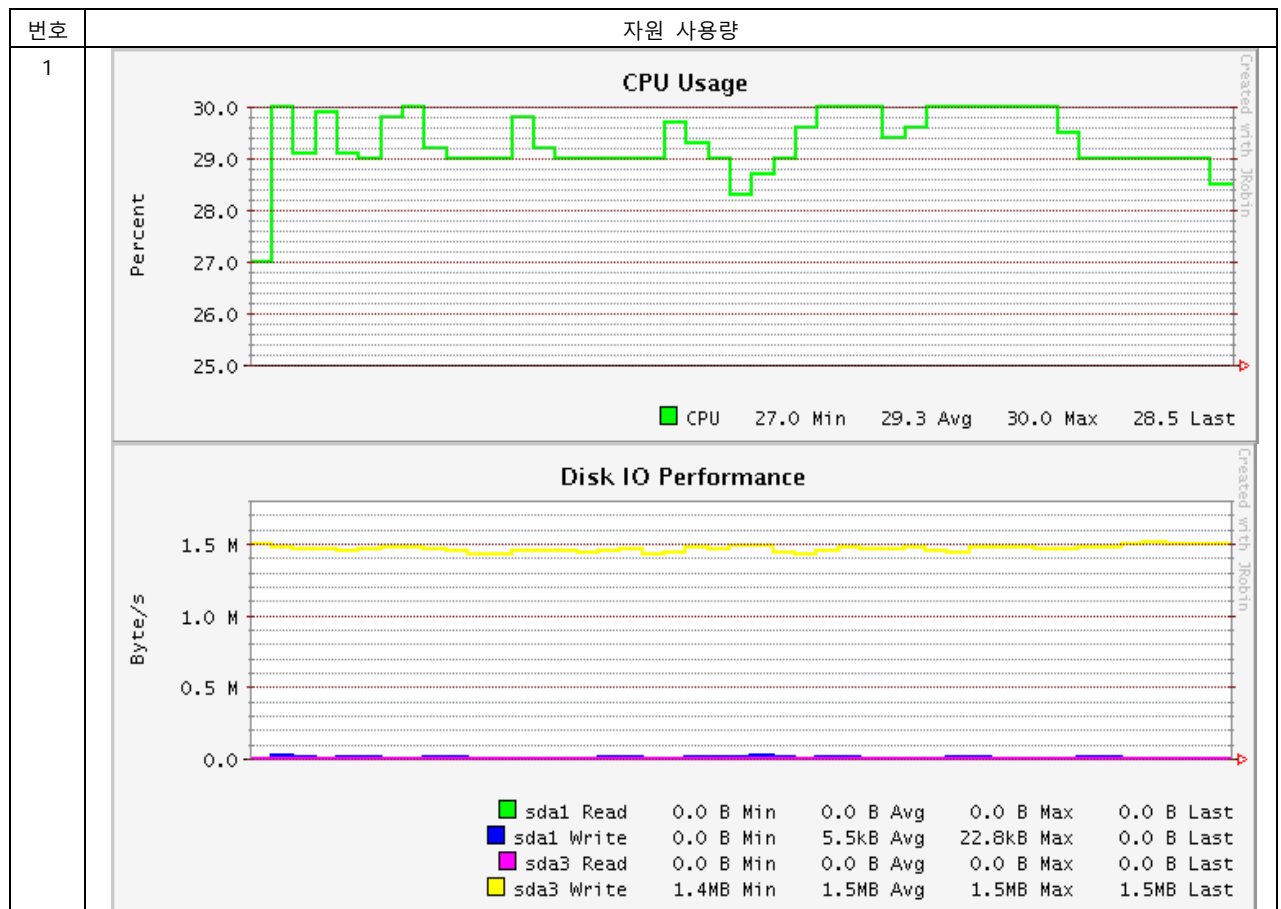
구분	측정값	비고
처리량	처리된 전체 서비스 요청수	Steady state 기간 동안 성공한서비스 요청수
	전체 실패한 서비스 요청수	Steady state 기간 동안 실패한 서비스 요청수
	초당 처리한 페이지 수 (PPS)	초당 처리된 서비스 요청 수
	초당 트랜잭션 수 (TPS)	1초 동안 오류없이 처리한 트랜잭션의 수
	초당 처리한 쿼리 수 (QPS)	1초 동안 expected response time을 만족하며 오류 없이 처리한 SQL수
응답시간	평균/최대/최소/90% 서비스 응답 시간	한 서비스의 시작부터 완료까지의 시간
	평균/최대/최소/90% 트랜잭션 응답 시간	트랜잭션의 시작부터 Commit/Rollback 작업 완료까지의 시간
	평균/최대/최소/90% 쿼리 응답 시간	하나의 쿼리 문이 처리되어 결과를 반환하기까지의 시간 (prepared statement의 경우 두번째 이후로는 execute만 측정)
자원 사용률	평균 CPU 사용량	측정 기간 동안 user /system/io wait를 합한 값 (100 ? CPU idle)
	평균 메모리 사용량	측정 기간 동안 사용된 메모리의 크기로 GB 단위로 측정
	평균 IO 사용량	측정 기간 동안 발생한 디스크 I/O의 양 (Bytes/Sec)과 수행시간(msec)

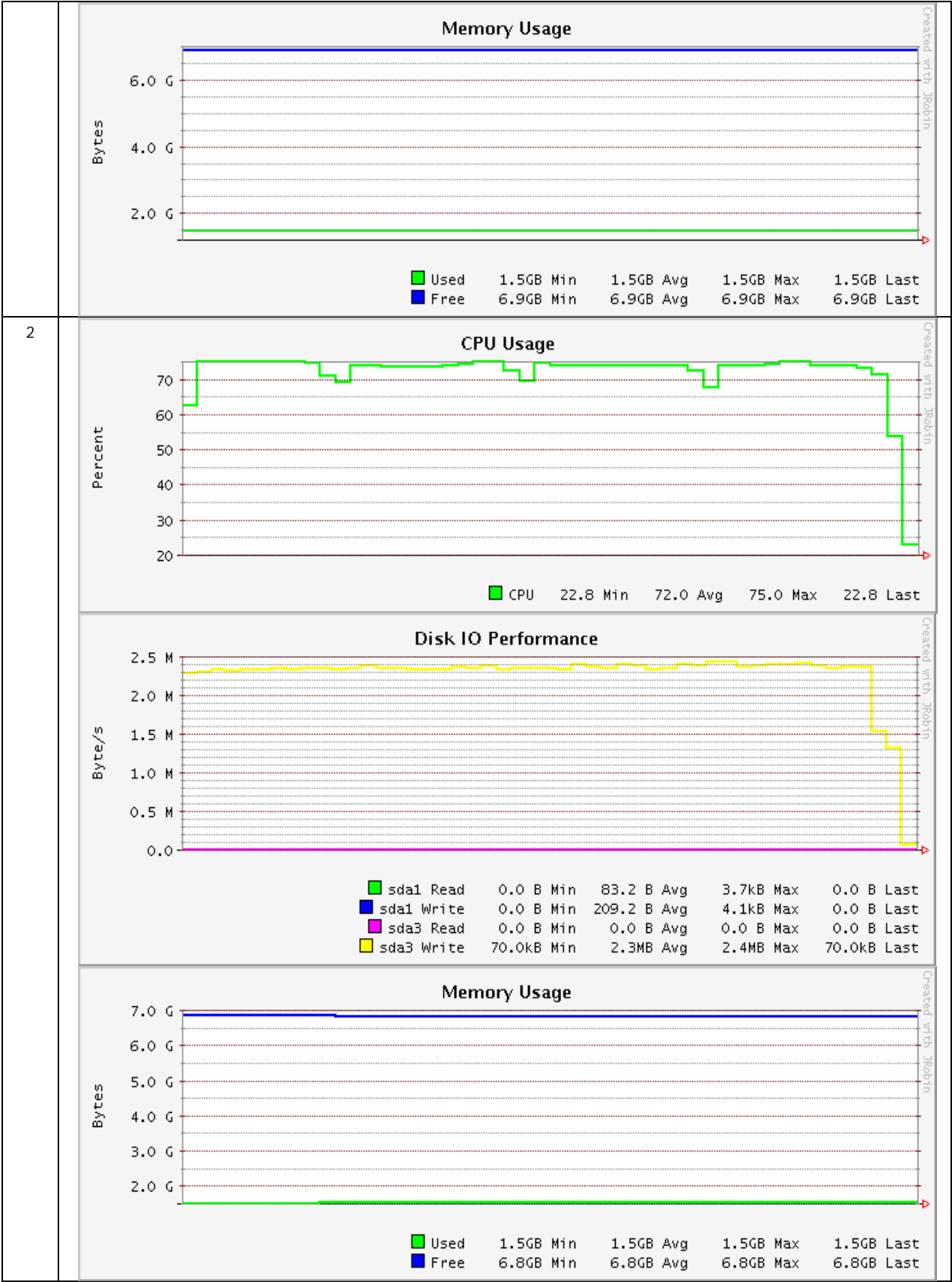
첨부 2: 벤치마크 시험 결과

<표 11> 시험 구성 및 시험 결과

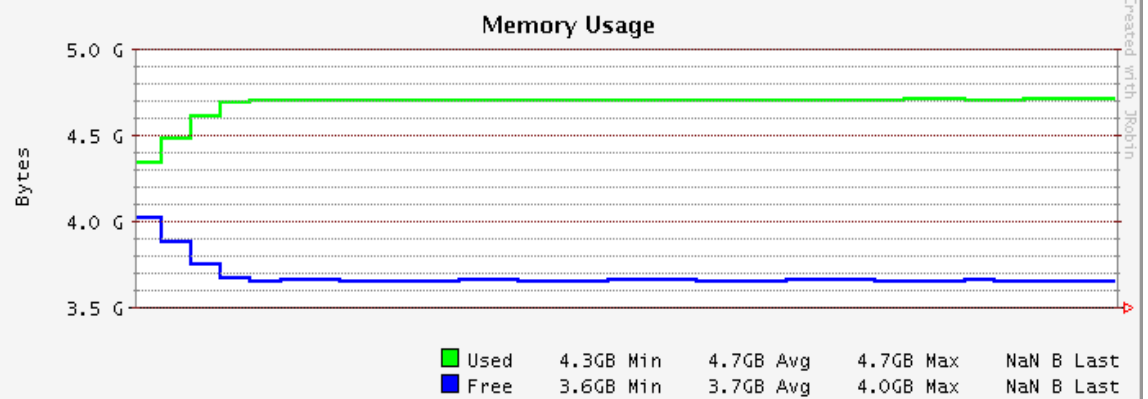
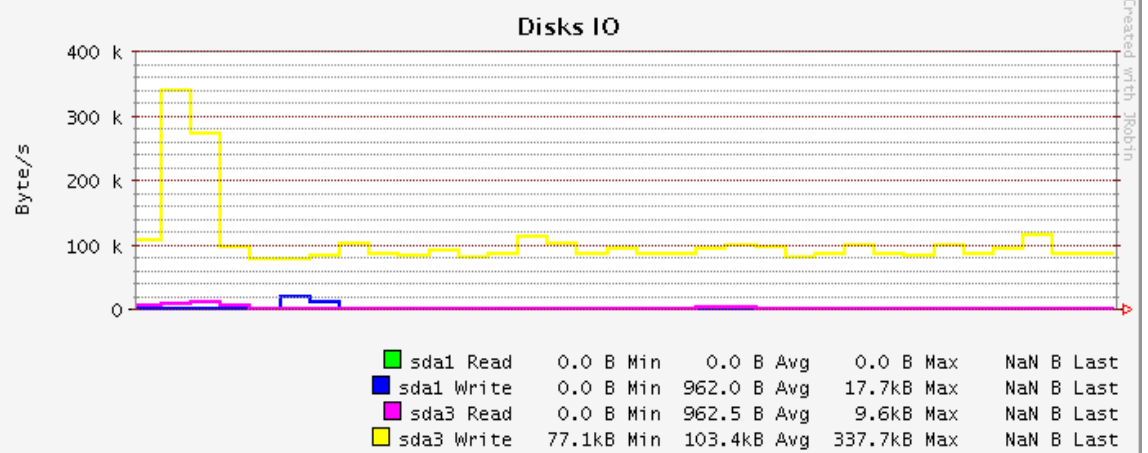
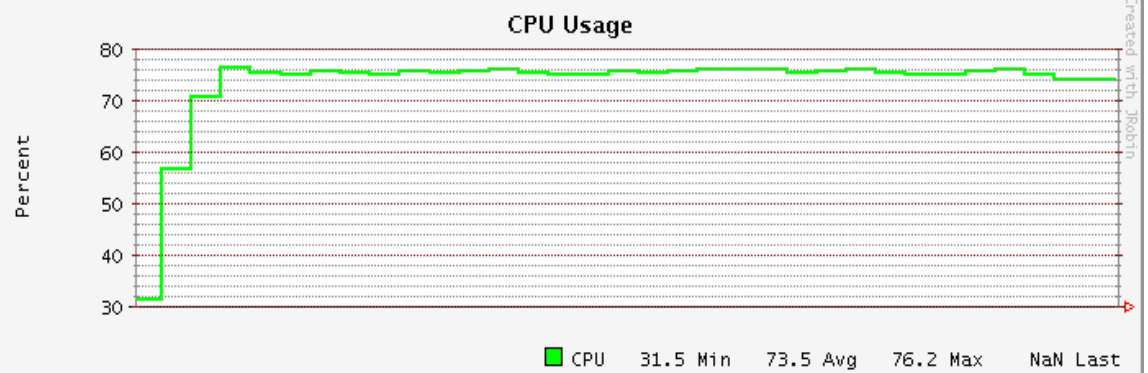
번호	DBMS	DB규모	Result Cache & JDBC Cache	#Page View	DB 서버의 CPU Usage
1	CUBRID	레벨 1	ON	1767	29.5%
2	CUBRID	레벨 2	ON	2121	75%
3	상용 DBMS D1	레벨 1	ON	2630	76%
4	상용 DBMS D1	레벨 2	ON	2928	83%
5	D2 DB	레벨 1	-	1184	100%
6	D2 DB	레벨 2	-	1195	100%
7	OSS DBMS D1	레벨 1	ON	593	29%
8	OSS DBMS D1	레벨 2	ON	387	18%

<표 12> 시험 별 리소스 사용량





3



4

