

[별첨 6]

Tomcat

테스트 결과보고서

2010. 11.

목 차

1. 테스트 대상 소개	1
2. 테스트 케이스 및 시나리오	1
가. 기능별 테스트 케이스 현황	1
나. 비 기능 테스트 시나리오	2
3. 기능 테스트 수행 결과	2
가. 기능 테스트 결과	2
나. 결함내역	2
다. 특이사항	2
4. 비 기능 테스트 수행 결과	3
가. 비 기능 테스트 결과	3
나. 비 기능 테스트 상세내역	3
5. 종합	9
참고자료	10

1. 테스트 대상 소개

가. Tomcat(v6.0.26)

아파치 톰캣(Apache Tomcat)은 아파치 소프트웨어 재단에서 개발된 서블릿 컨테이너(또는 웹 컨테이너)만 있는 웹 애플리케이션 서버이며, 자체적으로 HTTP 서버를 내장하고 있음

웹 서버와 연동하여 실행할 수 있는 자바 환경을 제공하여 자바 서버 페이지(JSP)와 자바 서블릿이 실행할 수 있는 환경을 제공함

관리 툴을 통해 설정을 변경할 수 있지만, XML 파일을 편집하여 설정할 수도 있음

아파치 톰캣은 Apache Licence, Version 2를 채용한 오픈소스 소프트웨어로써 서블릿이나 JSP를 실행하기 위한 서블릿 컨테이너를 제공하며, 상용 웹 애플리케이션 서버에서도 서블릿 컨테이너로 사용하는 경우가 많다. 버전 5.5이후는 기본적으로 Java SE 5.0 이후를 대응함

아파치 톰캣에 내장된 웹 서버로만 웹 시스템을 구성할 수 있지만, 대규모의 사용자가 사용하는 시스템을 구축하려면 웹 서버와 연동하는 안정적인 시스템을 구축해야 함

※ 출처 : 공개SW TRM - 솔루션 프로파일 “솔루션 설명” 참조

2. 테스트 케이스 및 시나리오

Tomcat의 신뢰성을 검증하기 위하여 테스트 케이스에 기반을 둔 기능 테스트와 테스트 시나리오에 기반을 둔 비 기능 테스트를 수행한다.

가. 기능별 테스트케이스 현황

[표 2-1. 기능별 테스트케이스 현황]

기 능	테스트 케이스 수
설치 및 삭제	3
실행	2
배포	2
로깅	3
보안	6
네이밍	3

모니터링	2
클러스터링	9
가상호스트	5
데이터베이스	5
기타(Default Servlet)	4
합 계	44

나. 비 기능 테스트 시나리오

[표 2-2. 비 기능 테스트 시나리오]

시나리오ID	설 명
OSS_WASR_01	가상사용자별 서버 응답시간 측정
OSS_WASR_02	프로세스(쓰레드) 개수에 따른 서버 성능 측정
OSS_WASR_03	JDBC Datasource DBCP 개수에 따른 성능 측정
OSS_WASR_04	12시간 장시간 테스트를 진행하여 서버의 안정성 측정

3. 기능 테스트 결과

기능 테스트 수행관련 세부 절차 및 결과는 별첨 「Tomcat 테스트 케이스」를 참고한다.

가. 기능 테스트 결과

[표 3-1. 기능 테스트 결과]

테스트케이스	Pass	Fail	Not Available
44	41	0	3

나. 결함내역

☐ 테스트 수행 중 치명적인 결함은 발견되지 않음

다. 특이사항

☐ 제품 매뉴얼에서 삭제 관련 정보가 제공되지 않음

4. 비 기능 테스트 수행 및 결과

가. 비 기능 테스트 결과

[표 4-1. 비 기능 테스트 수행 결과]

테스트 시나리오	내용	결과
OSS_WASR_01	가상사용자별 응답시간 측정	상세내역 참조
OSS_WASR_02	프로세스(쓰레드) 수에 따른 성능 측정	프로세스 수 300에서 최적
OSS_WASR_03	DBCP 개수에 따른 성능 측정	DBCP 수 100에서 최적
OSS_WASR_04	장시간 수행을 통한 안정성 테스트	총 2,135,282번 수행

나. 비 기능 테스트 상세내역

비 기능 테스트의 경우 하드웨어 사양뿐 아니라, OS 및 애플리케이션 구성에 따라 성능 측정 결과가 상이하므로, 실제 운영 환경에서 적용할 경우 테스트 결과가 다를 수 있다.

※ 용어설명

가상사용자 : 테스트 툴에서 생성한 쓰레드 수(사용자)

Average : 테스트 평균 응답 시간

Deviation : 각각의 응답시간과 평균 응답시간과의 차이의 합

Throughput : 서버가 처리하는 작업량

□ OSS_WASR_01 - 가상사용자별 서버 응답시간을 측정한다.

○ 테스트 조건

- 최대 쓰레드 개수 : 200개(default)
- 가상사용자 : 1000명, 1500명, 2000명
- 웹페이지 크기 : 100KB
- 응답시간 제한 없음

○ 테스트 절차

- JMeter 툴을 실행하여 가상사용자 1000, 1500, 2000명 생성 후 테스트 대상 페이지에 접속
 - Number of Threads : 1000, 1500, 2000명, Ramp-Up : 0, Loop Count : 1

○ 테스트 결과

- Summary Report

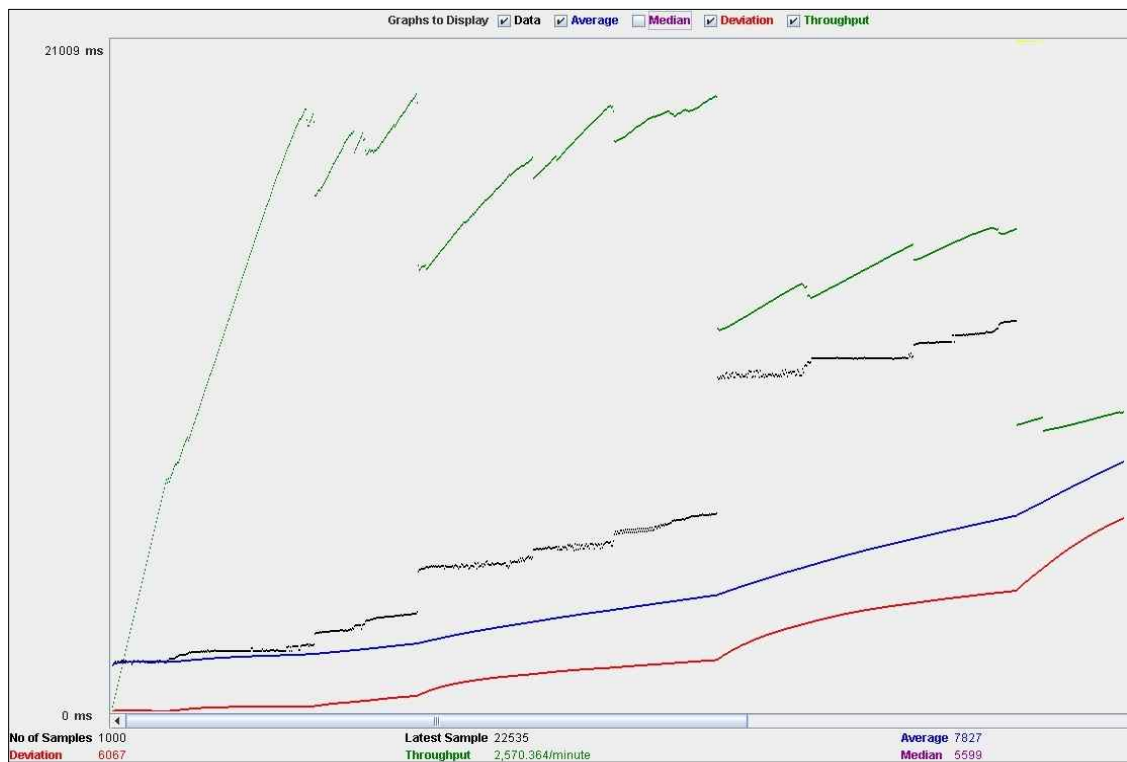
[표 4-2. 가상사용자별 응답시간]

가상사용자	평균시간/ms	최소시간/ms	최대시간/ms	에러율	TPS	KB/sec
1000	7827	1458	22578	0.00%	43	4031
1500	12289	1459	22563	0.00%	65.4	4143
2000	14536	1426	23087	0.00%	85.2	4404

- 가상사용자가 증가함에 따라 평균응답 시간이 점진적으로 증가하지만, TPS 수치는 오히려 증가하여, 사용자 증가에 따른 서버의 작업 처리능력은 증가하였음

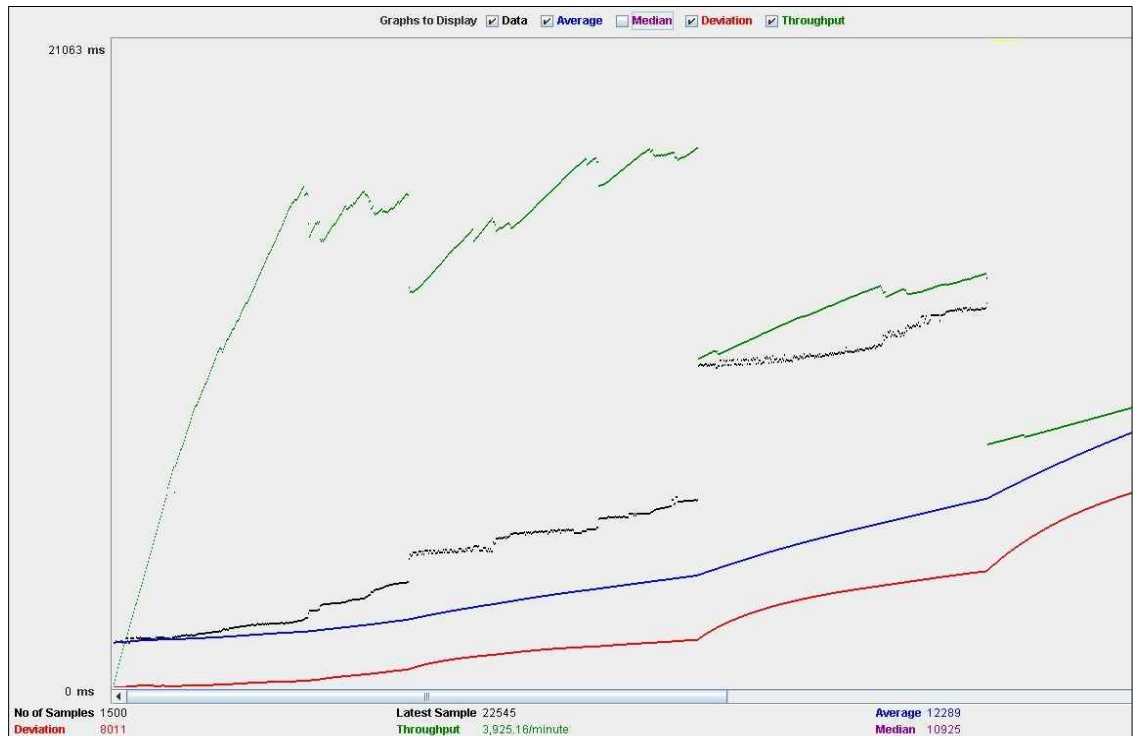
- Graph Results

- 가상사용자 1000명



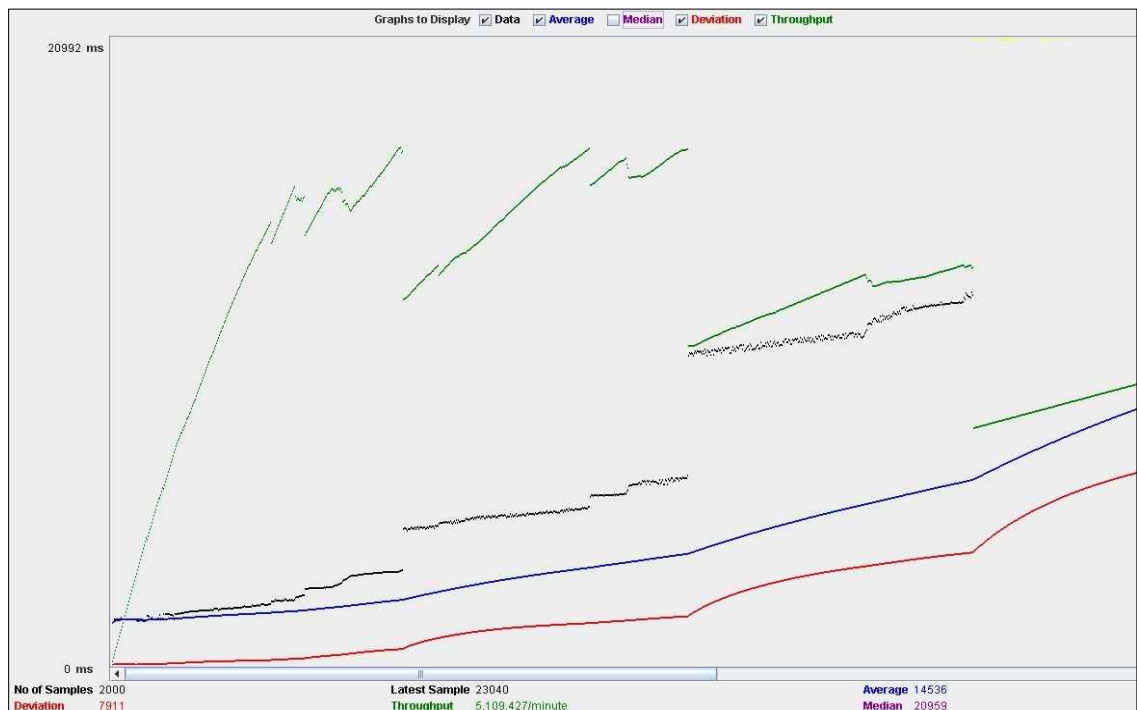
[그림4-1. 가상사용자 1000명]

▪ 가상사용자 1500명



[그림4-2. 가상사용자 1500명]

▪ 가상사용자 2000명



[그림4-3. 가상사용자 2000명]

□ OSS_WASR_02 - 프로세스(쓰레드) 개수에 따른 성능을 측정한다.

○ 테스트 조건

- 쓰레드 수 : 200(default), 250, 300
- 가상사용자 : 1000명

○ 테스트 절차

- JMeter 툴을 실행하여 가상사용자 1000명 생성 후 테스트 대상 페이지에 접속
 - Number of Threads : 1000명, Ramp-Up : 0, Loop Count : 1
- 환경설정파일에서 프로세스 수를 조건에 맞게 변경 후 재시작하여 테스트

○ 테스트 결과

- Summary Report

[표 4-4. 프로세스(쓰레드)별 응답시간]

쓰레드 수	평균시간/ms	최소시간/ms	최대시간/ms	에러율	TPS	KB/sec
200	7827	1458	22578	0.00%	43	4031
250	6512	1675	12927	0.00%	76.3	7365
300	6163	1817	12404	0.00%	77.3	7461

- 쓰레드 수가 증가함에 따라 평균응답 시간이 다소 빨라졌으며, TPS 수치도 오히려 증가하여 서버의 작업 처리능력은 증가하였으나, 이 결과가 무조건 쓰레드 수를 증가시킨다고 성능이 향상된다는 것을 의미하는 것은 아니며, 하드웨어 성능 및 서버 구성에 따라 달라질 수 있음

□ OSS_WASR_03 - JDBC Datasource DBCP 개수에 따른 성능을 측정한다.

○ 테스트 조건

- DBCP 개수 : 10(default), 50, 100
- 가상사용자 : 1000명
- 테스트 데이터 : 10만건

○ 테스트 절차

- JMeter 툴을 실행하여 가상사용자 1000명 생성 후 테스트 대상 페이지에 접속
 - Number of Threads : 1000명, Ramp-Up : 0, Loop Count : 1
- 테스트 페이지에서 테스트 데이터 10만건 중 100건의 데이터를 질의하여 출력

○ 테스트 결과

- Summary Report

[표 4-5. DBCP 개수별 응답시간]

DBCP 수	평균시간/ms	최소시간/ms	최대시간/ms	에러율	TPS	KB/sec
10	81	2	185	0.00%	786.2	3381
50	131	2	2963	0.00%	329.3	1416
100	38	2	750	0.00%	999	4296

- DBCP의 수가 50에서 100으로 증가할 경우 평균응답 시간은 단축되고, TPS 수치는 증가하여 서버의 처리량이 전반적으로 증가하였음
- DBCP의 수가 너무 크게 설정 할 경우 메모리의 자원을 많이 소비 할 가능성이 있으므로, 주의를 요함

□ OSS_WASR_04 - 12시간 장시간 테스트를 진행하여 안정성을 측정한다.

○ 테스트 조건

- 12시간 연속 서버로 테스트 페이지 요청
- 가상사용자 : 50명

○ 테스트 절차

- 테스트 수행 전에 nmon 실행
 - nmon -f -t -r tomcatTest -s60 -c750 (1분마다 실행하며, 750분간 동작)
- JMeter 툴을 실행하여 가상사용자 50명 생성 후 테스트 대상 페이지에 접속
 - Number of Threads : 50명, Ramp-Up : 60, Loop Count : forever
 - Scheduler : 12시간 실행되도록 설정

○ 테스트 결과

- Summary Report

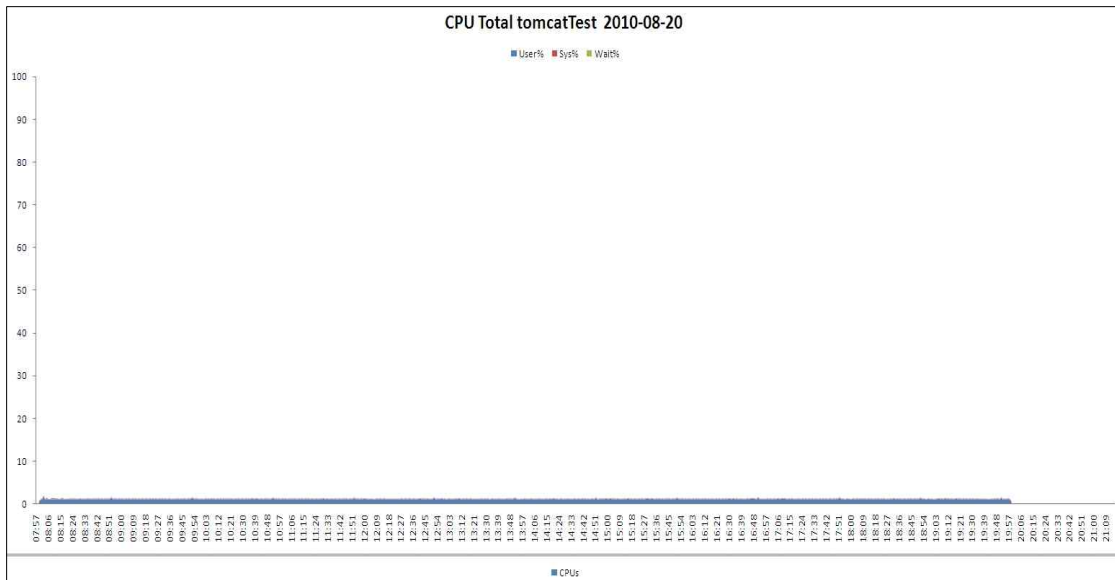
[표 4-6. 12시간 테스트 결과]

총실행건수	평균시간/ms	최소시간/ms	최대시간/ms	에러율	TPS	KB/sec
2135282	1011	1004	4040	0.00%	49.4	4773

- 12시간 동안 수행되면서 총 2,135,282번 수행되었으며, 최대응답시간과 최소응답시간의 차이가 크지 않고, 평균응답 시간이 1초 정도로 에러 없이 장시간동안 안정적으로 수행되었음

- 서버 자원 현황

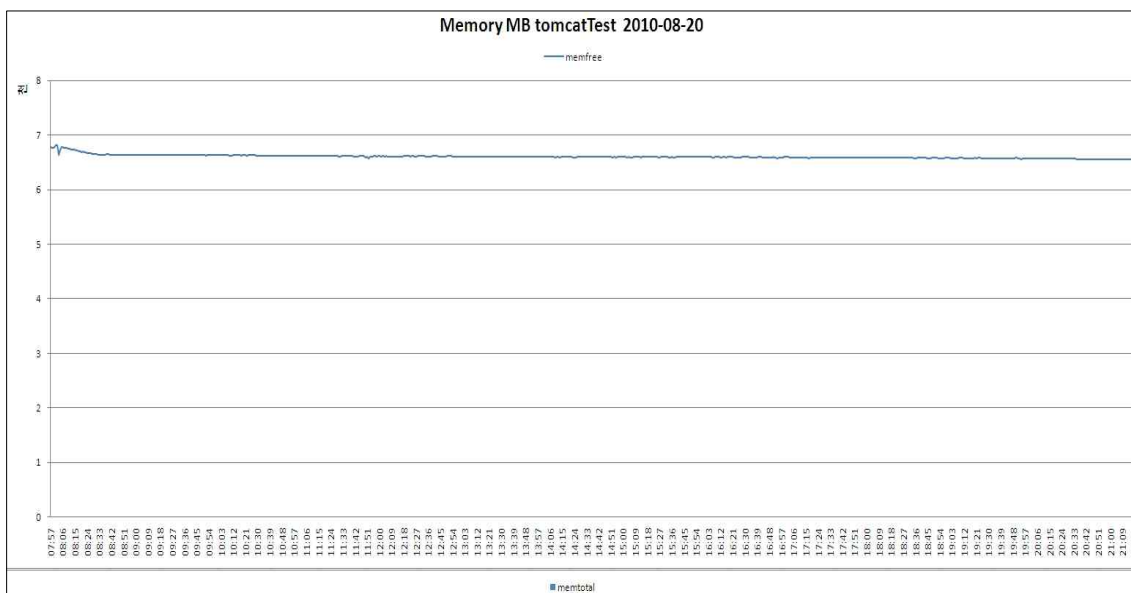
▪ CPU 사용률



[그림4-4. CPU 사용률]

12시간 수행되면서 cpu 점유율이 50% 이하로 형성되는 안정적인 모습을 보이고 있음

▪ 메모리 사용량



[그림4-5. 메모리 사용량]

메모리 사용량 변화가 일정하게 형성되는 안정적인 모습을 보이고 있음

5. 종합

- ☐ 테스트 케이스 기반 기능 테스트에 대한 테스트 결과 치명적인 결함이 발생하지 않고 정상적으로 동작함
- ☐ 비 기능 테스트에 대한 테스트 결과 및 여러 가지 상황의 테스트 조건에 대한 테스트 결과 치명적인 결함은 발생하지 않았으며, 전반적인 서버 자원 사용률도 안정적으로 사용함

※ 참고 자료

[1] <http://tomcat.apache.org/>

[2] <http://jakarta.apache.org/jmeter/>

[3]

https://h10078.www1.hp.com/cda/hpms/display/main/hpms_content.jsp?zn=bto&cp=1-11-126-17%5E8_4000_100

[4] <http://nmon.sourceforge.net/>

[5] <http://www.ibm.com/developerworks/wikis/display/WikiPtype/nmonanalyser>