
분산DB 분야 Stack 통합 테스트 결과보고서 [Cassandra]

2012. 07.

목 차

I. Stack 통합 테스트 개요	3
1. 목적	3
II. 분산DB 소개	4
1. 분산DB	4
2. 분산DB 분야 주요 공개SW	6
III. 테스트 대상 소개	7
1. Cassandra 소개	7
IV. Stack 통합 테스트	8
1. 테스트 환경	8
2. 주요 테스트 방법	9
3. 테스트 시나리오 현황	10
4. 기능 테스트 수행 결과	10
5. 성능 테스트 수행 결과	12
V. 종합	37
※ 참고자료	38

[별첨 1] Cassandra 테스트 케이스

[별첨 2] Cassandra 테스트 환경구축 가이드

I. Stack 통합 테스트 개요

공개SW Stack 통합테스트는 여러 공개SW들의 조합으로 시스템 Stack을 구성한 후 Stack을 구성하는 공개SW의 상호운용성에 중점을 두고 기능 및 성능테스트 시나리오를 개발하여 테스트를 진행한다.

본 통합테스트를 통해 안정된 Stack 정보를 제공하여 민간 및 공공 정보시스템 도입 시 활용될 수 있도록 한다.

1. 목적

- 공개SW Stack 통합 테스트 수행 목적
 - 공개SW로 구성된 Stack이 유기적으로 잘 동작함을 확인
 - 다양한 Stack 구성에 기반을 둔 테스트를 통해 안정된 Stack 조합 규명
 - 공개SW 시스템 도입을 위한 Stack 참조모델의 신뢰성 정보로 활용
 - 공개SW의 신뢰성과 범용성에 대한 사용자 인식 제고

II. 분산DB 소개

1. 분산 DB(NoSQL)

□ 분산 DB(NoSQL) 개요

NoSQL은 관계형 데이터베이스 관리 시스템(RDBMS)과 다른 새로운 형태의 데이터베이스 관리 시스템 클래스이다. NoSQL과 같은 데이터 저장소는 고정된 테이블 스키마를 필요로 하지 않으며, Join 연산이 없고, 수평적 확장을 지원한다.

NoSQL 구조는 eventual consistency¹⁾ 같은 결과적 일관성을 보장하며 트랜잭션은 1개의 항목에 대해서만 지원한다. 하지만 일부 시스템에서는 AppScale²⁾이나 CloudTPS³⁾같은 보조 미들웨어를 통해서 ACID⁴⁾ 기능을 보장한다.

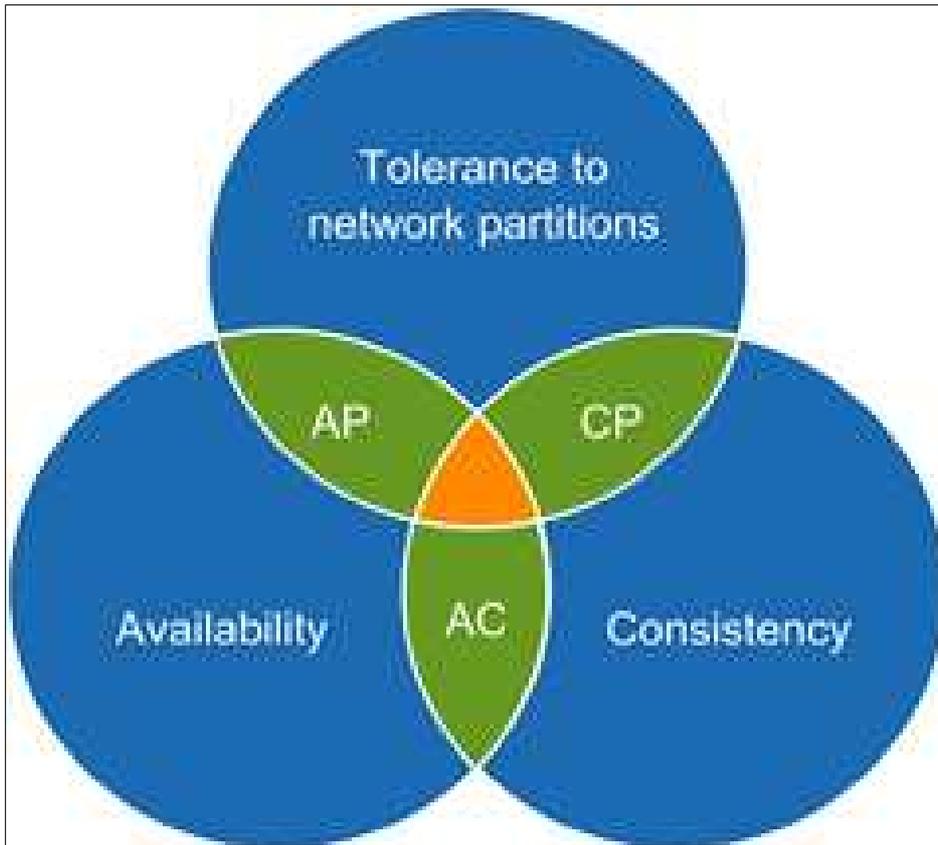
NoSQL 시스템은 여러 서버에 이중화 방식의 분산 구조를 사용하며 데이터는 분산 해시 테이블을 이용하여 여러 대의 서버에 중복되어 저장된다. 이러한 방식으로 시스템은 필요 시 서버를 추가하는 방식으로 확장 가능하다.

일부 NoSQL은 연관배열(associative arrays) 혹은 키 값(key-value) 쌍 같은 단순한 인터페이스를 선호한다. 반면 XML 데이터베이스 같은 시스템에서는 XQuery 표준을 지원한다. CloudTPS 같은 시스템은 Join 쿼리를 지원하기도 한다. (출처 : WIKIPEDIA)

-
- 1) 특정 시점에는 일관성이 결여되는 경우가 발생하지만, 일정 시간이 경과 후에는 일관성을 유지함
 - 2) IaaS 클라우드에서 Google App Engine 애플리케이션을 쉽게 실행할 수 있게 하는 클라우드 컴퓨팅 플랫폼
 - 3) 클라우드 웹 응용 프로그램을 위한 확장 가능한 트랜잭션
 - 4) 데이터베이스 트랜잭션이 안전하게 수행된다는 것을 보장하기 위한 성질을 가리키는 약어로 원자성, 일관성, 독립성, 지속성을 지칭한다.

□ CAP 정리

CAP 정리에 따르면, 대규모 분산데이터시스템의 속성을 3가지로 구분 하였으며, 최소 2가지를 만족해야한다.



[그림 II-1. CAP 이론 다이어그램]

- 일관성(Consistency)
모든 노드는 동시에 같은 데이터를 참조해야 함
- 가용성(Availability)
모든 요청이 성공하거나, 실패에 상관없이 응답을 보장해야 함
- 지속성(Partition tolerance)
일부 데이터를 손실하더라도 사용자는 사용이 가능해야 함

2. 분산DB 분야 주요 공개SW

[표 II-1. 주요 공개SW]

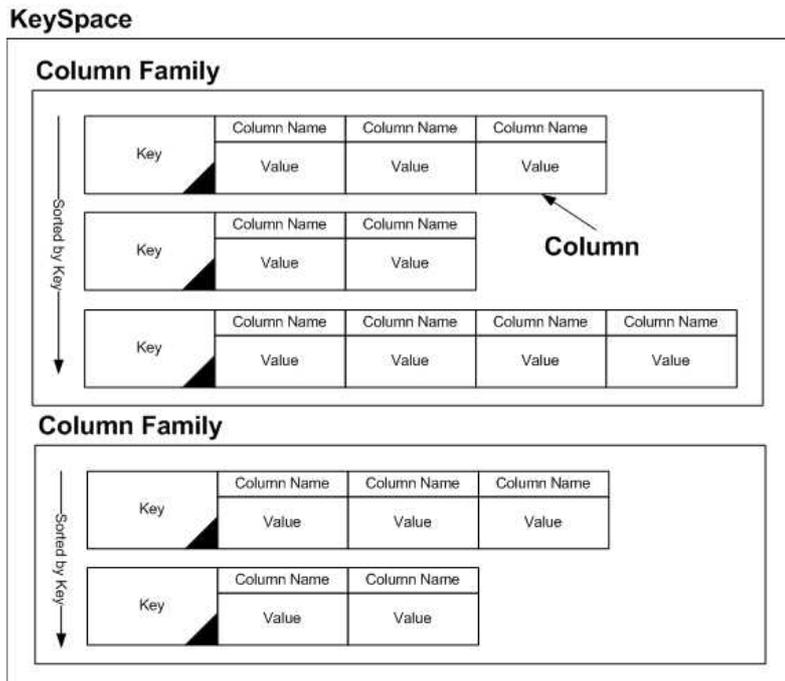
제품명	Stack 환경		홈페이지	비고
Cassandra	OS	Cross-platform	http://cassandra.apache.org	Apache License 2
	JAVA	JDK 6		
Cloudata	OS	Linux	http://www.cloudata.org	Apache License 2
	JAVA	JDK 6		
CouchDB	OS	Cross-platform	http://couchdb.apache.org	Apache License 2
HBase	OS	Cross-platform	http://hbase.apache.org	Apache License 2
	JAVA	JDK 6		
MongoDB	OS	Cross-platform	http://www.mongodb.org	GNU AGPL v3.0

III. 테스트 대상 소개

1. Cassandra 소개

Cassandra는 확장성이 높고 결과적 일관성(eventually consistent) 특징을 가지고 있으며, 분산 환경을 지원하는 컬럼 기반 분산 데이터베이스 시스템이다.

Cassandra는 분산시스템 모델인 Dynamo와 구글의 BigTable 데이터 모델을 기반으로 개발되었으며, 페이스북에서 2008년에 공개SW로 전환되었으며, 최근에는 트위터 및 페이스북 등 소셜 네트워크 서비스(SNS) 관련 데이터를 처리하는 용도로 사용 중이다.



[그림 III-1. Cassandra Data Model]

IV. Stack 통합 테스트

1. 테스트 환경

Cassandra 환경

[표 IV-1. Cassandra 환경]

모듈	Version
Cassandra	1.1.1
SimpleCassie (PHP Library)	0.7.1.6
JDK	1.7.0_04

Stack 환경

[표 IV-2. Stack 환경]

구성	OS	비고
A Stack	CentOS 6.2 (64bit)	Master (X3850M2)
	CentOS 6.2 (64bit)	Slave (DL360G6)
B Stack	Windows Server 2008 R2 (64bit)	Master (X3850M2)
	Windows Server 2008 R2 (64bit)	Slave (DL360G6)

HW 환경

[표 IV-3. HW 환경]

제조사	모델명	CPU	MEM	Disk	NIC
IBM	X3850M2	2.40GHz (Six Core) x 4, 12MB L3 90w	8GB	146G x 4	Gigabit
HP	DL360G6	2.40 GHz (Quad-Core) x 2	8GB	146G x 5	Gigabit

2. 주요 테스트 방법

□ 시나리오 테스트

시나리오 테스트 기법은 단일 기능에 대한 결함 여부를 확인하는 것이 아니라 서로 다른 컴포넌트 사이의 상호작용과 간섭으로 발생할 수 있는 결함을 발견하기 위한 기법이다.

본 테스트에서는 사용자 시나리오 테스트 기법을 적용하여 Cassandra를 사용하는 사용자들이 사용할 수 있는 항목 중 Cassandra-Cli(명령어 기반 클라이언트)를 이용한 데이터 조작, cqlsh(쿼리 도구), 클러스터, 복제, 모니터링 등에 대한 사용자 시나리오 도출하였으며, 각각의 항목에서 도출한 시나리오 중 사용자가 일반적으로 수행할 수 있는 시나리오를 추출하여 테스트케이스를 작성하였다.

□ 상호 운용성 기반 테스트

애플리케이션이 지원하는 다양한 Stack을 구성하여, 각각의 서버 환경에서 결함이 없이 잘 구동되는지를 확인하며, 비즈니스 로직 및 데이터 상호 연동이 Stack 내의 컴포넌트 간의 인터페이스가 정상적으로 동작하는지 여부를 중점적으로 검증한다.

3. 테스트 시나리오 현황

[표 IV-4. 테스트 시나리오 현황]

기능	테스트 시나리오	테스트 케이스
Cassandra-Cli	6	34
cqlsh	5	16
Cluster	4	13
Replication	4	17
Backup	2	10
Logging	3	8
Monitoring	3	22
합 계	27	120

4. 기능 테스트 수행 결과

테스트 수행 관련 세부 시나리오 및 테스트 결과는 별첨 「Cassandra 테스트 케이스」 문서를 참고한다.

□ 테스트 결과

[표 IV-5. 테스트 결과]

분류		PASS		FAIL		N/A	
기능	개수	A Stack	B Stack	A Stack	B Stack	A Stack	B Stack
Cassandra-Cli	34	33	33	1	1	0	0
cqlsh	16	16	0	0	0	0	16
Cluster	13	13		0	0	0	0
Replication	17	17	17	0	0	0	0
Backup	10	10	10	0	0	0	0
Logging	8	8	8	0	0	0	0
Monitoring	22	22	22	0	0	0	0

□ 결합내용

o Secondary Index 데이터 조회 결합

- Column Key를 제외한 데이터에 대한 조회를 하기 위해서 해당 Column Name에 대한 Secondary Index를 설정해야 정상적으로 데이터 조회가 가능함
- A, B Stack에서 동일하게 Secondary Index를 설정 후 데이터 조회가 되지 않는 동일한 결합 발생함
- Cassandra BTS(Bug Tracking System) 내용 확인결과 이전 버전에서 문제가 발생하여, 결합이 수정된 것으로 확인되나, 정확한 원인 해결을 위한 추가적인 파악이 필요함

□ 참고사항

- o cqlsh(Cassandra SQL-like Command)의 경우 윈도우 환경에서는 지원이 되지 않는 기능이므로, B Stack에서는 수행하지 않음

5. 성능 테스트 수행 결과

성능 테스트의 경우 하드웨어 사양뿐 아니라, OS 및 애플리케이션 환경 구성에 따라 성능 측정 결과가 상이하므로, 실제 운영 시스템 환경에 따라 테스트 결과가 다를 수 있다.

본 성능 테스트는 웹서비스(SimpleCassie 활용⁵⁾)를 이용하여, Cassandra 데이터 조회 및 입력에 대한 시간반응성 및 자원효율성을 측정하였음

□ 테스트 시나리오

[표 IV-6. 테스트 시나리오]

시나리오 ID	시나리오
SC_CSD_1	단일서버 순차 입력(50만건)
SC_CSD_2	단일서버 랜덤 입력
SC_CSD_3	단일서버 랜덤 조회
SC_CSD_4	클러스터 ⁶⁾ 순차 입력(50만건)
SC_CSD_5	클러스터 랜덤 입력
SC_CSD_6	클러스터 랜덤 조회

□ 서버 설정 정보

[표 IV-7. 서버 설정 정보]

구분		항목
WEB	PHP	memory_limit = 512M max_input_time = 60 max_execution_time = 30
	HTTP	MaxClients 150 KeepAliveTimeout 5 Timeout 300
Cassandra DB (Java Heap Memory)		A Stack : 2G B Stack : 1G
Network Bandwidth	PC<->WEB	94Mbits/sec
	WEB<->DB	941 Mbits/sec

5) Cassandra DB와 연동하기 위한 PHP Library

6) 다중 머신에 분산하여 구성 후 데이터를 각각의 서버에 분산하여 부하 관리

□ 측정항목

[표 IV-8. 측정항목]

항목	내용
TPS	단위시간 당 트랜잭션 처리 수
응답시간	사용자 요청을 처리하기 위한 소요된 총 시간
자원사용률	CPU, Memory 등 서버 자원사용 현황

□ 테스트 결과

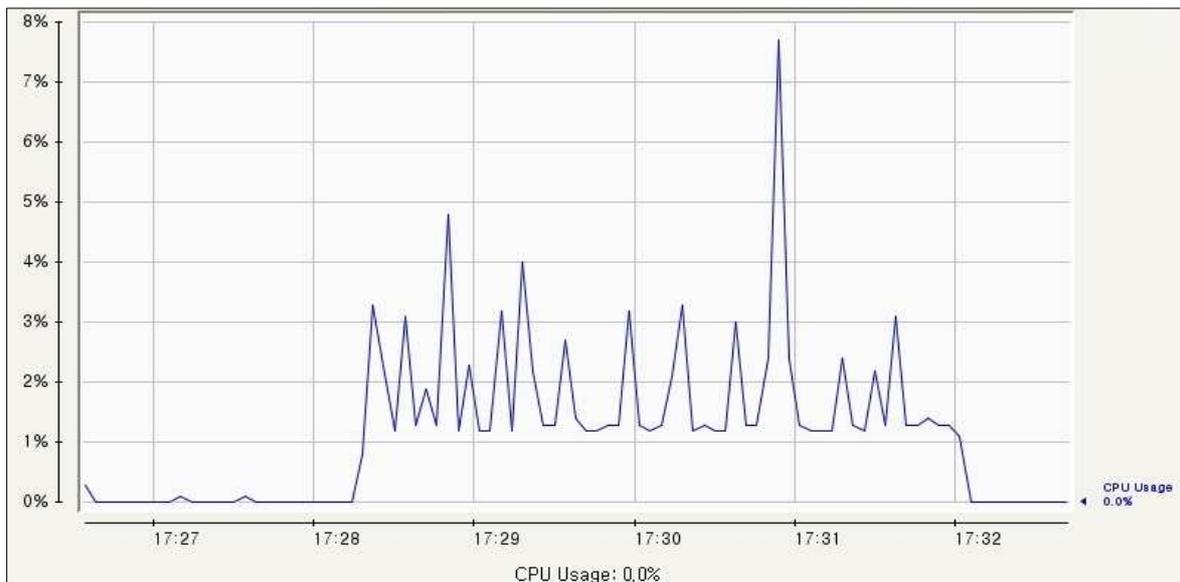
o 단일서버 순차 입력(50만건)

- 수행정보

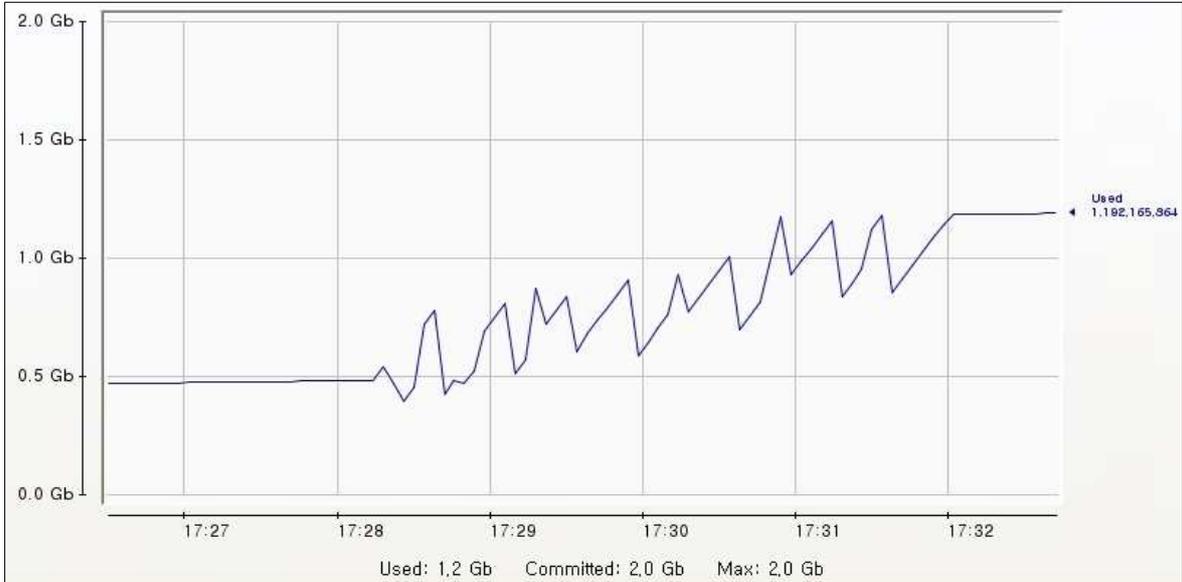
수행 조건	PHP에서 50만건 데이터 입력
-------	-------------------

- 측정결과 (A Stack)

- 데이터 입력 소요시간 : 224초



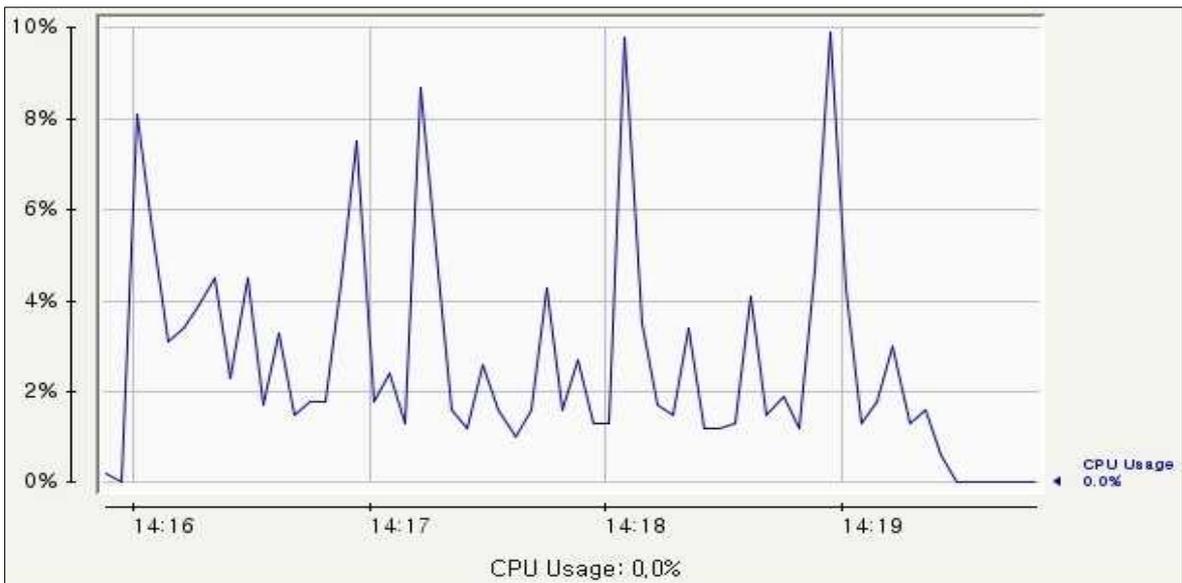
[그림 IV-1. CPU 자원사용률]



[그림 IV-2. Memory 자원사용률]

- 측정결과 (B Stack)

- 데이터 입력 소요시간 : 206초



[그림 IV-3. CPU 자원사용률]



[그림 IV-4. Memory 자원사용률]

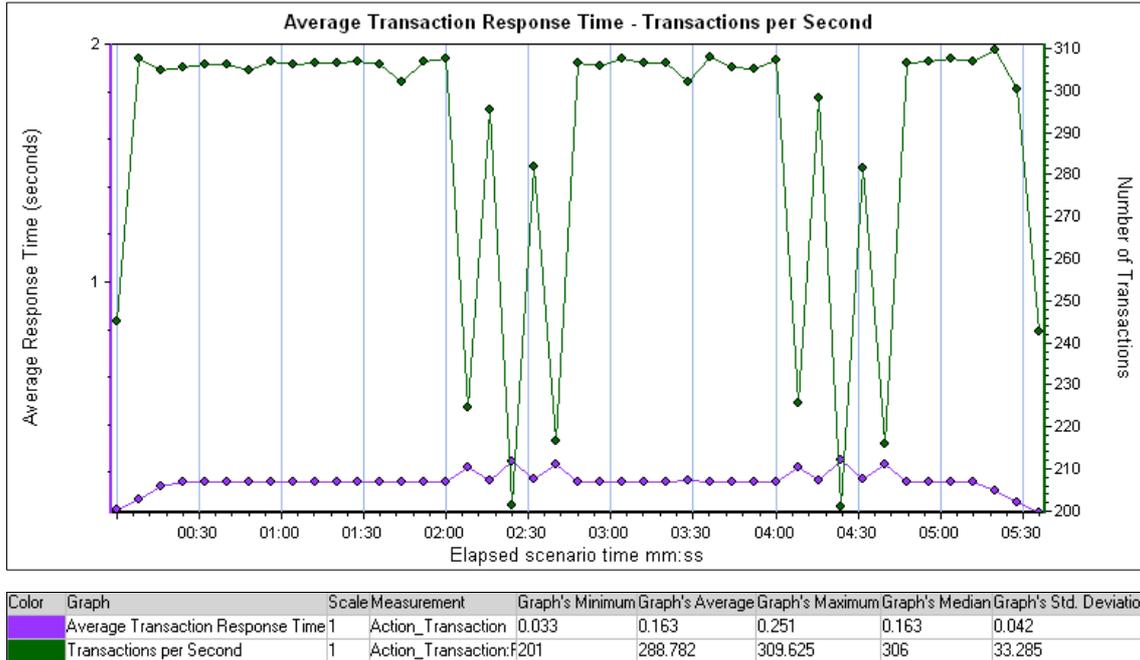
o 단일서버 랜덤 입력

- 수행정보

부하발생	<ul style="list-style-type: none"> - Lamp up : 5User / 1초 - Running-Time : 5분 - Lamp down : 5User / 1초 - 동시사용자 : 50User
수행 조건	<ul style="list-style-type: none"> - UUID⁷⁾를 생성하여 중복되지 않은 키 생성 및 데이터 입력

7) Universally Unique Identifier의 약자로 128비트 고유한 데이터 값

- 측정결과 (A Stack)

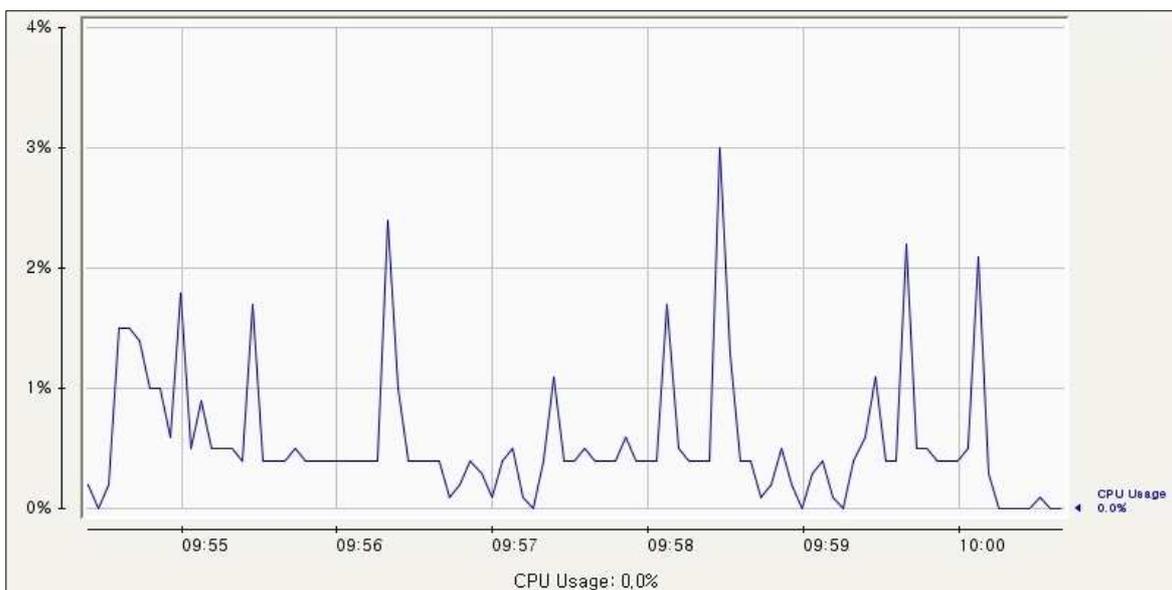


[그림 IV-5. 트랜잭션 결과]

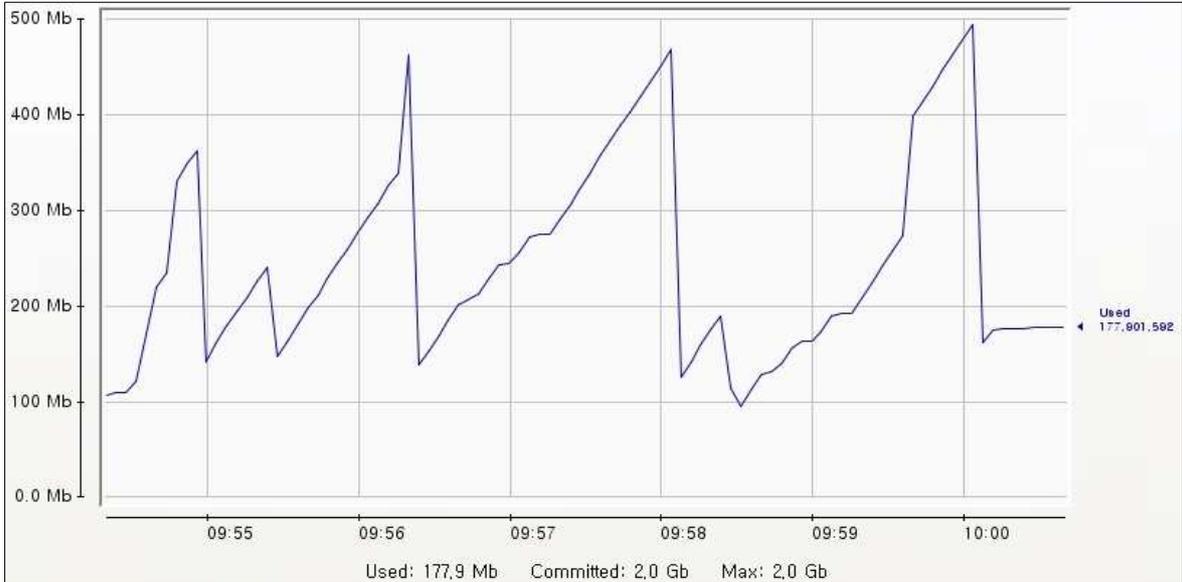
응답시간은 평균 1초 이내의 안정적인 응답을 보이고 있음

[참고사항]

(공통) 트랜잭션 처리가 주기적으로 잠시 저하되는 원인은 메모리에 캐쉬된 데이터를 디스크에 쓰는 과정 중에 발생하는 현상으로 판단됨

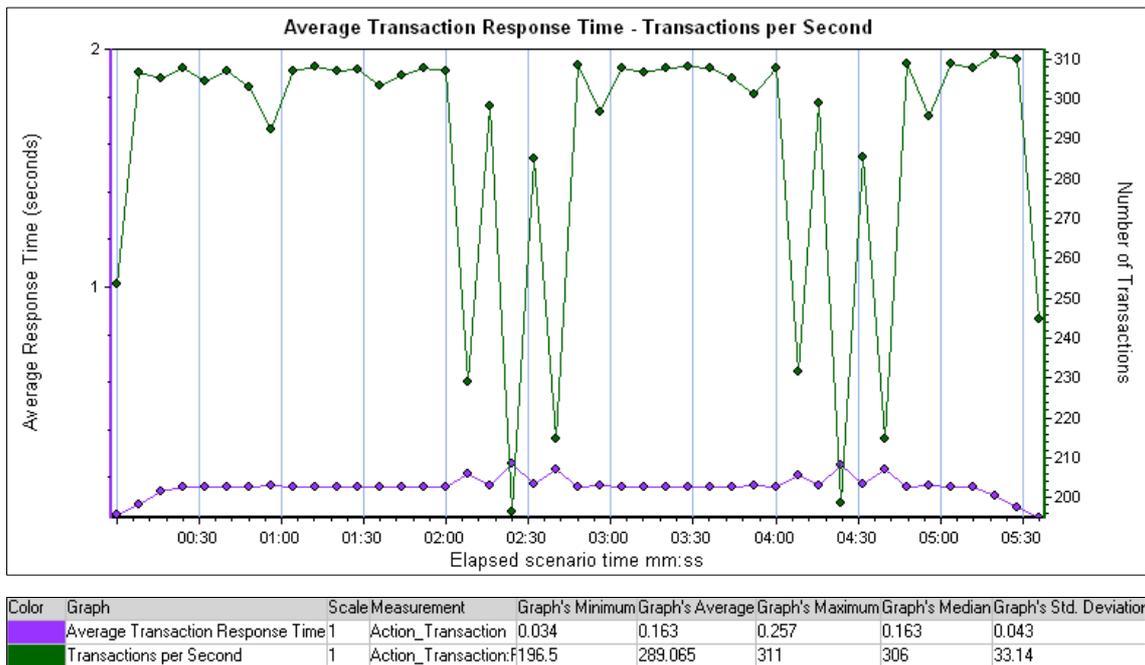


[그림 IV-6. CPU 자원사용률]



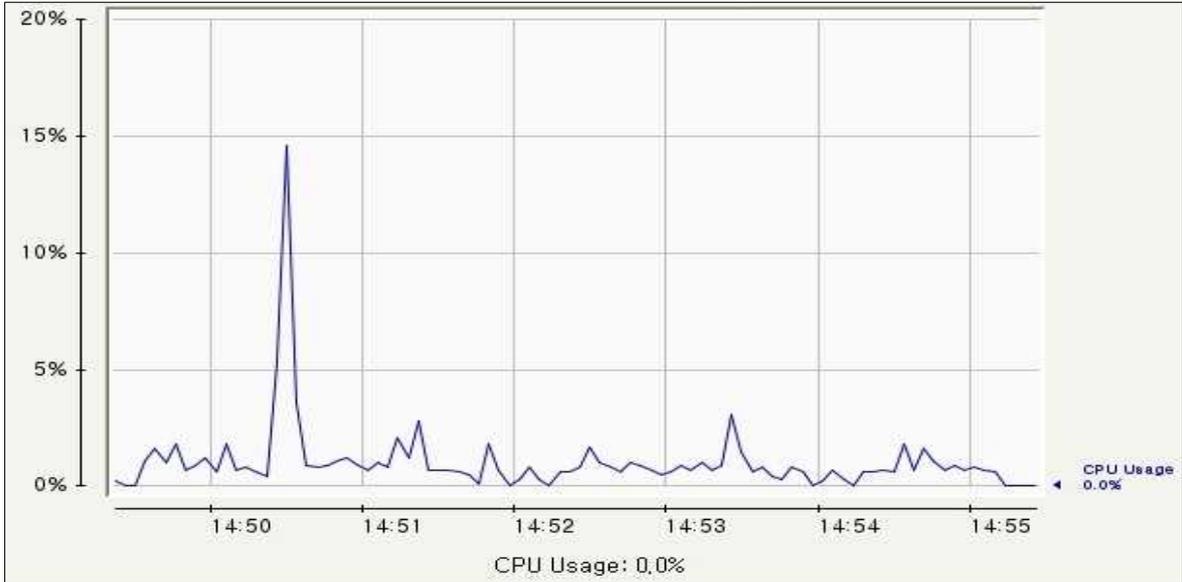
[그림 IV-7. Memory 자원사용률]

- 측정결과 (B Stack)



[그림 IV-8. 트랜잭션 결과]

응답시간은 평균 1초 이내의 안정적인 응답을 보이고 있음



[그림 IV-9. CPU 자원사용률]



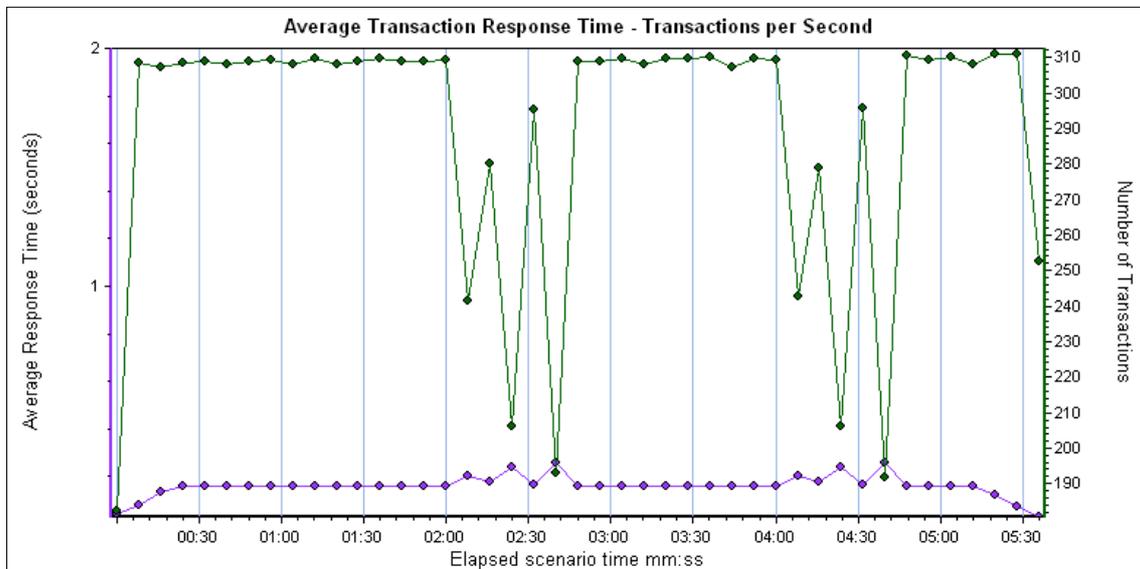
[그림 IV-10. Memory 자원사용률]

o 단일서버 랜덤 조회

- 수행정보

부하발생	<ul style="list-style-type: none"> - Lamp up : 5User / 1초 - Running-Time : 5분 - Lamp down : 5User / 1초 - 동시사용자 : 50User
수행 조건	<ul style="list-style-type: none"> - CF(Column Family)에 등록 된 키 값을 랜덤하게 생성한 후 데이터 조회 - \$bbs_key = rand(1, 500000);

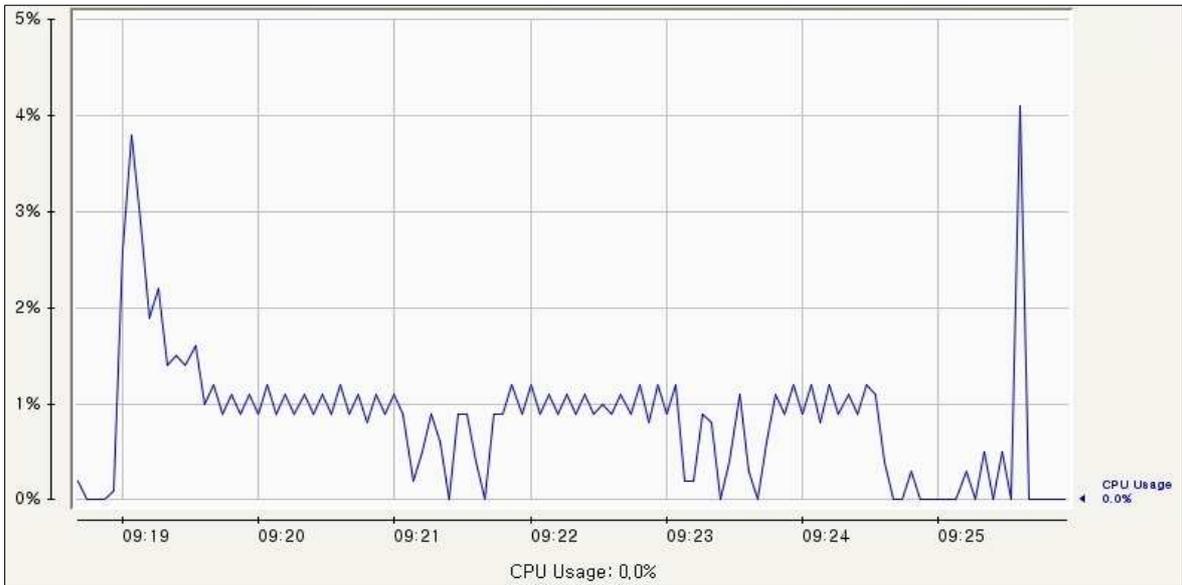
- 측정결과 (A Stack)



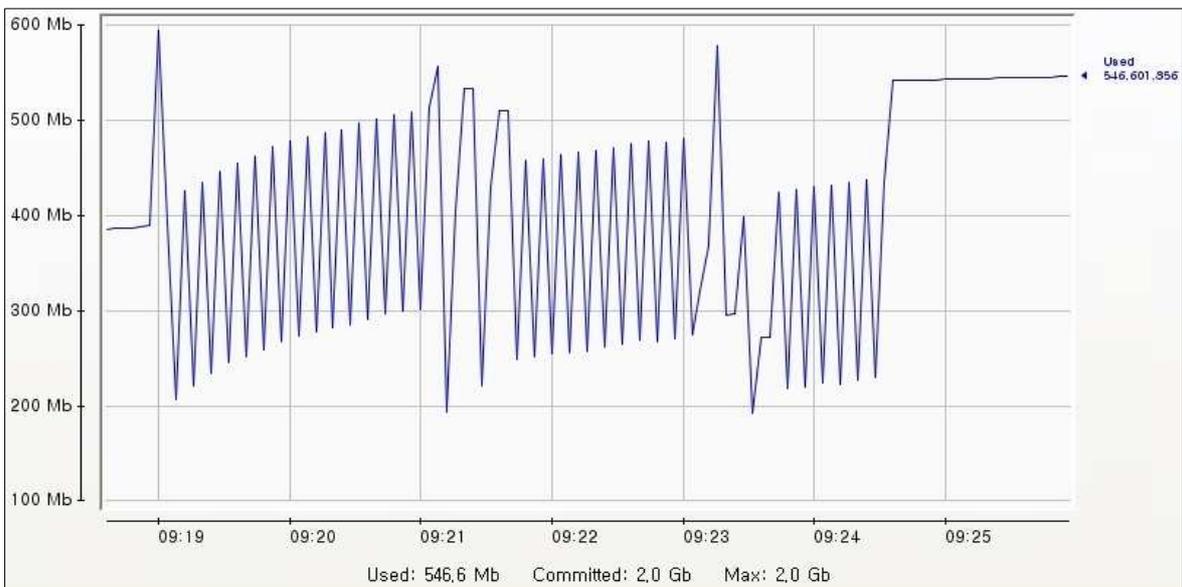
Color	Graph	Scale	Measurement	Graph's Minimum	Graph's Average	Graph's Maximum	Graph's Median	Graph's Std. Deviation
	Average Transaction Response Time	1	Action_Transaction	0.034	0.162	0.259	0.162	0.043
	Transactions per Second	1	Action_Transaction:F	182.375	289.523	311	308.75	38.087

[그림 IV-11. 트랜잭션 결과]

응답시간은 평균 1초 이내의 안정적인 응답을 보이고 있음

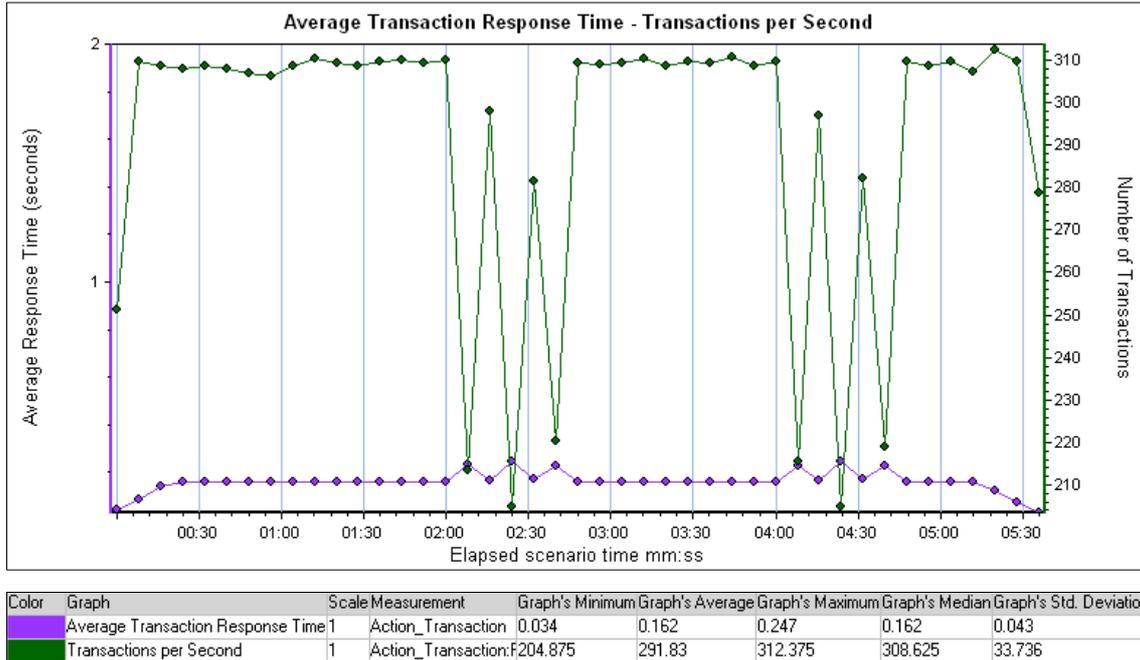


[그림 IV-12. CPU 자원사용률]



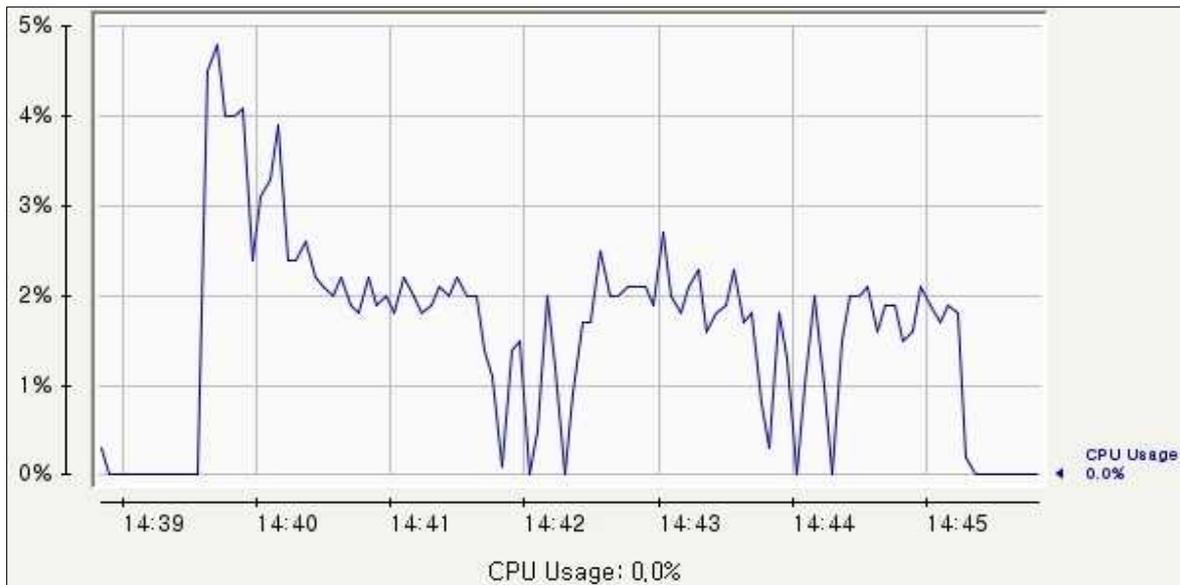
[그림 IV-13. Memory 자원사용률]

- 측정결과 (B Stack)

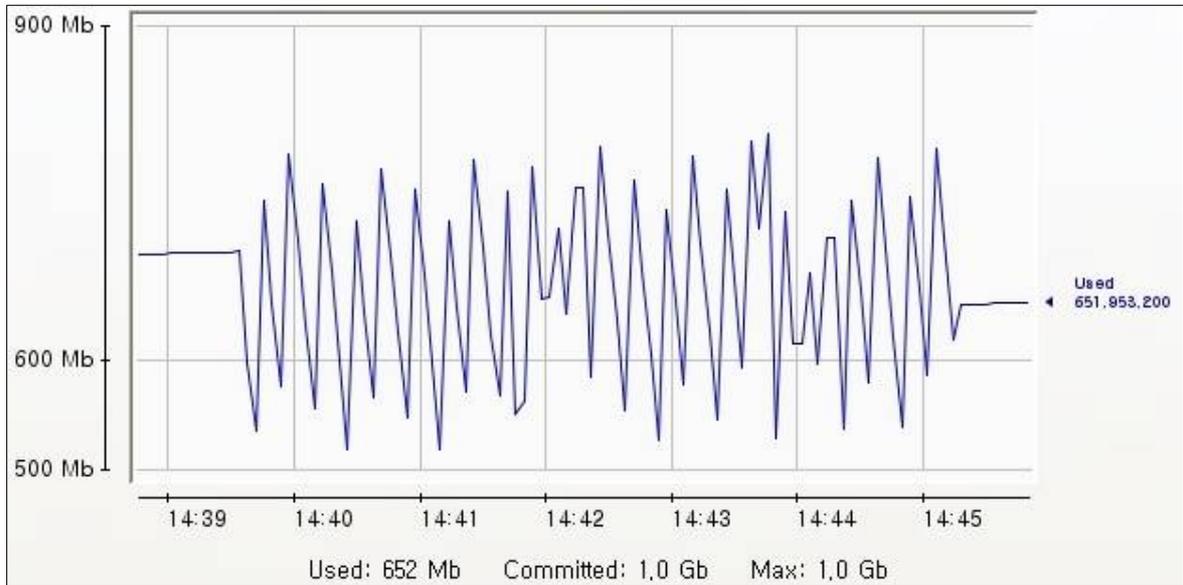


[그림 IV-14. 트랜잭션 결과]

응답시간은 평균 1초 이내의 안정적인 응답을 보이고 있음



[그림 IV-15. CPU 자원사용률]



[그림 IV-16. Memory 자원사용률]

o 클러스터 순차 입력(50만건)

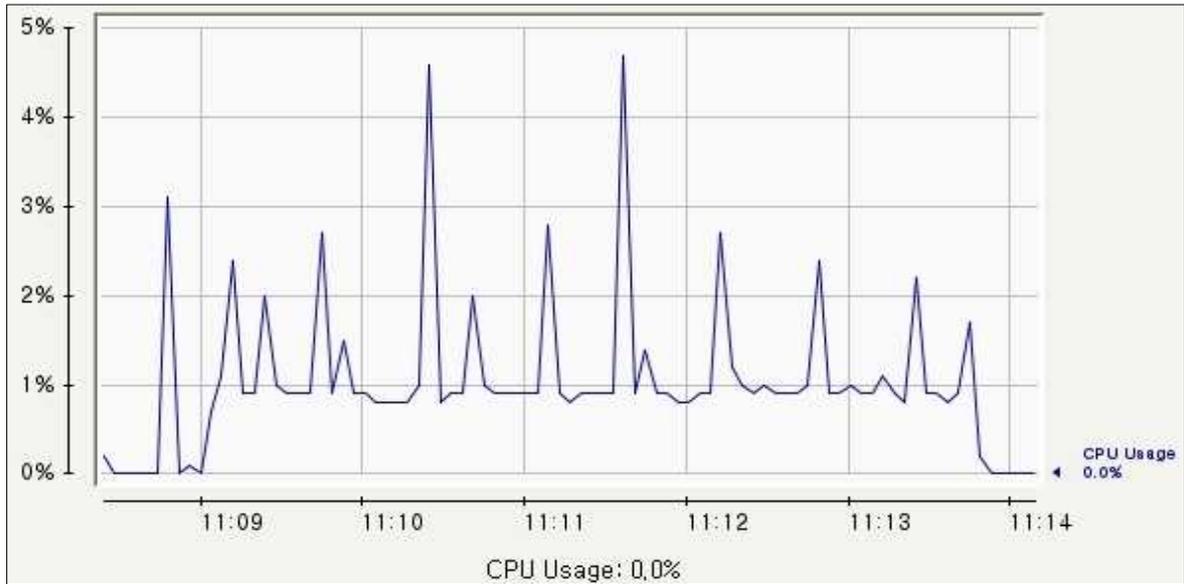
- 수행정보

수행 조건	PHP에서 50만건 데이터 입력
-------	-------------------

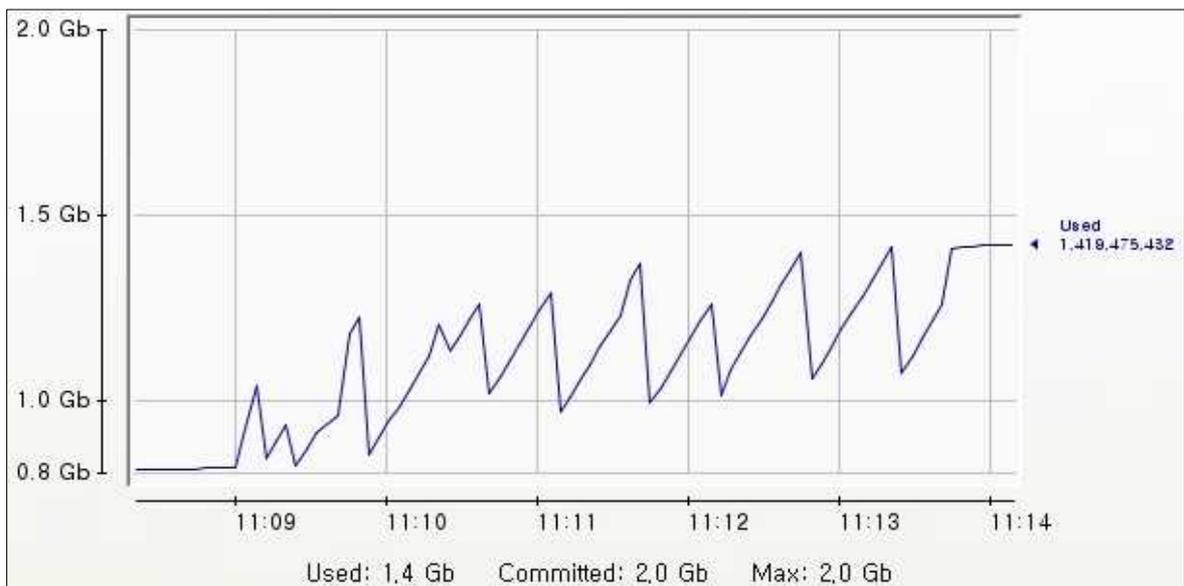
- 측정결과 (A Stack)

- 데이터 입력 소요시간 : 279초

- 클러스터 A노드 (Seed서버)

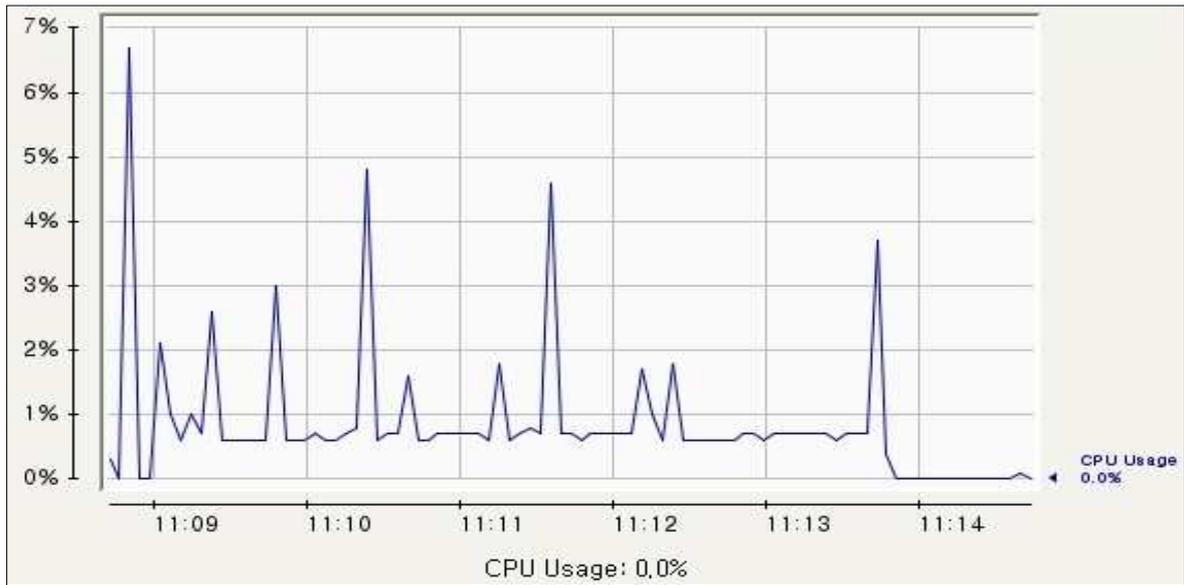


[그림 IV-17. CPU 자원사용률]

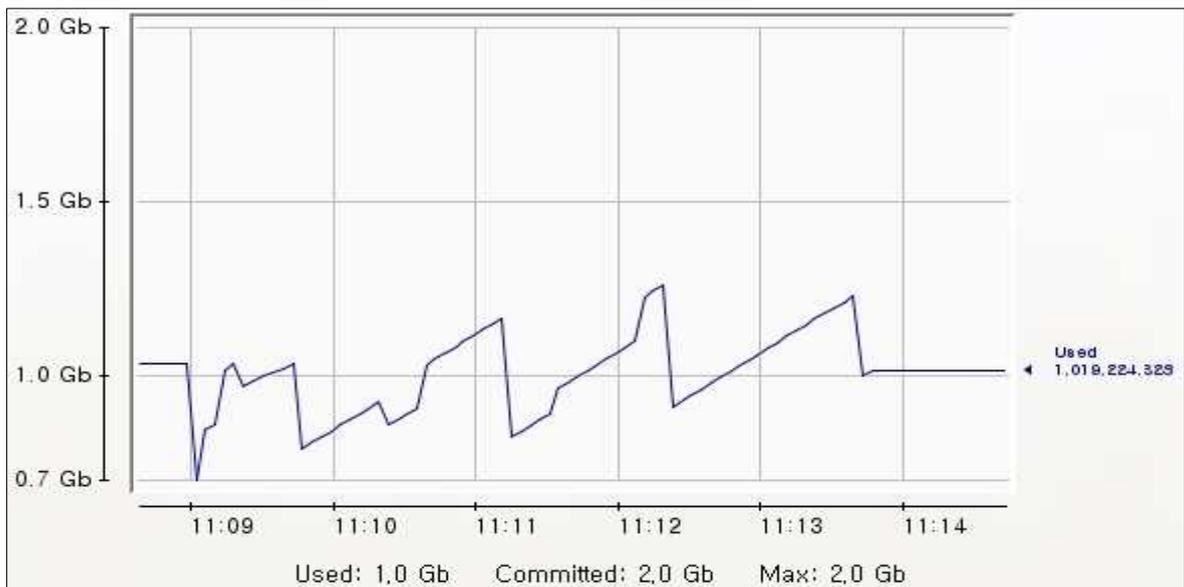


[그림 IV-18. Memory 자원사용률]

- 클러스터 B노드



[그림 IV-19. CPU 자원사용률]

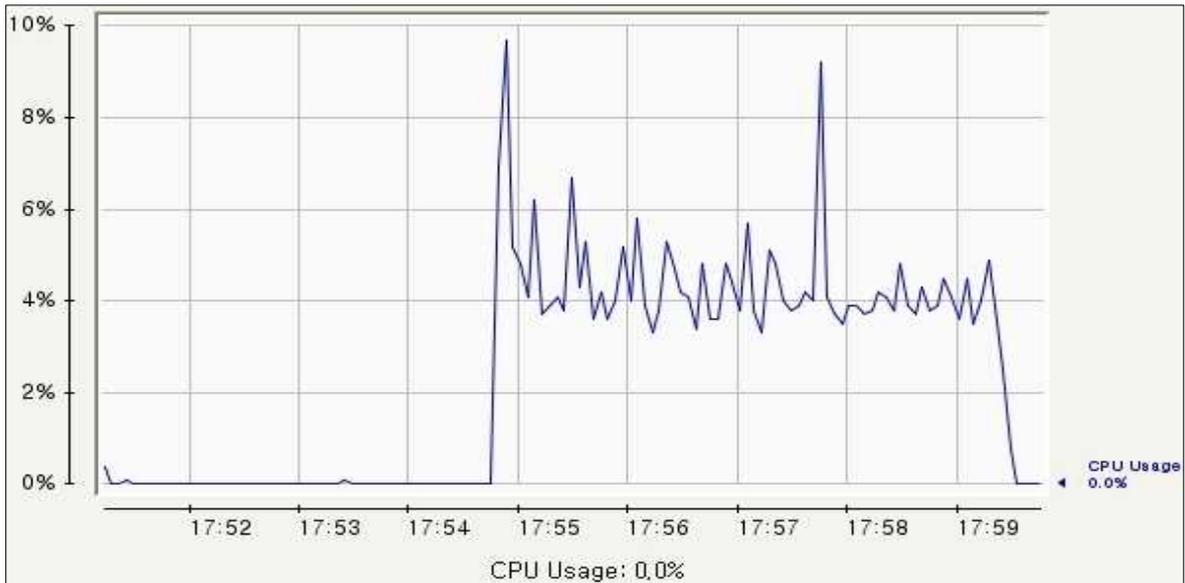


[그림 IV-20. Memory 자원사용률]

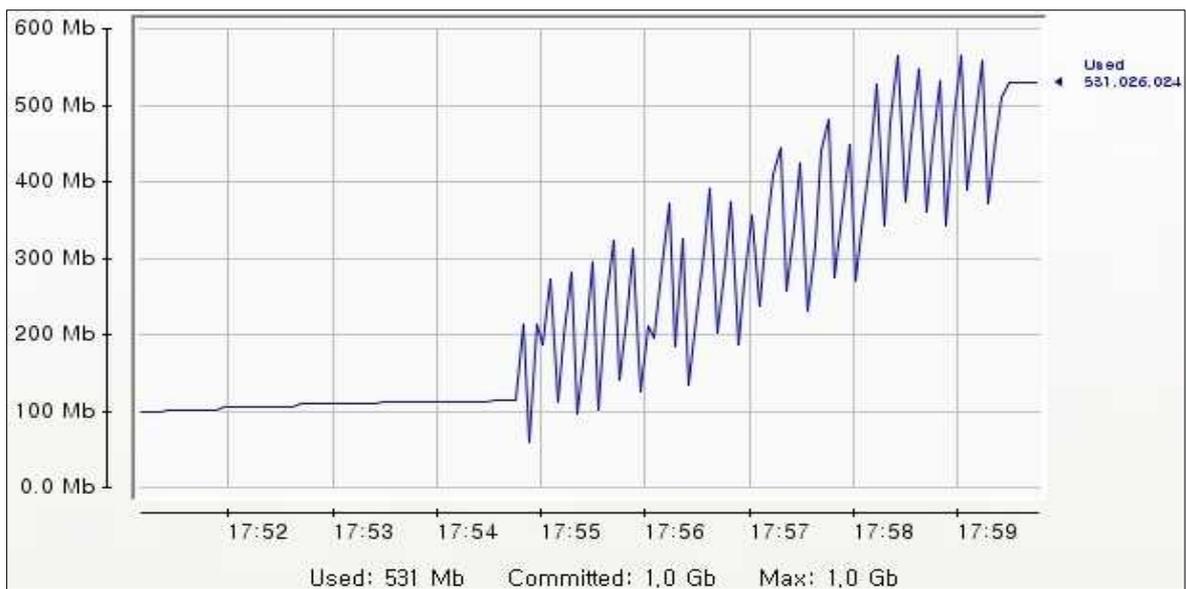
- 측정결과 (B Stack)

- 데이터 입력 소요시간 : 281초

- 클러스터 A노드 (Seed서버)

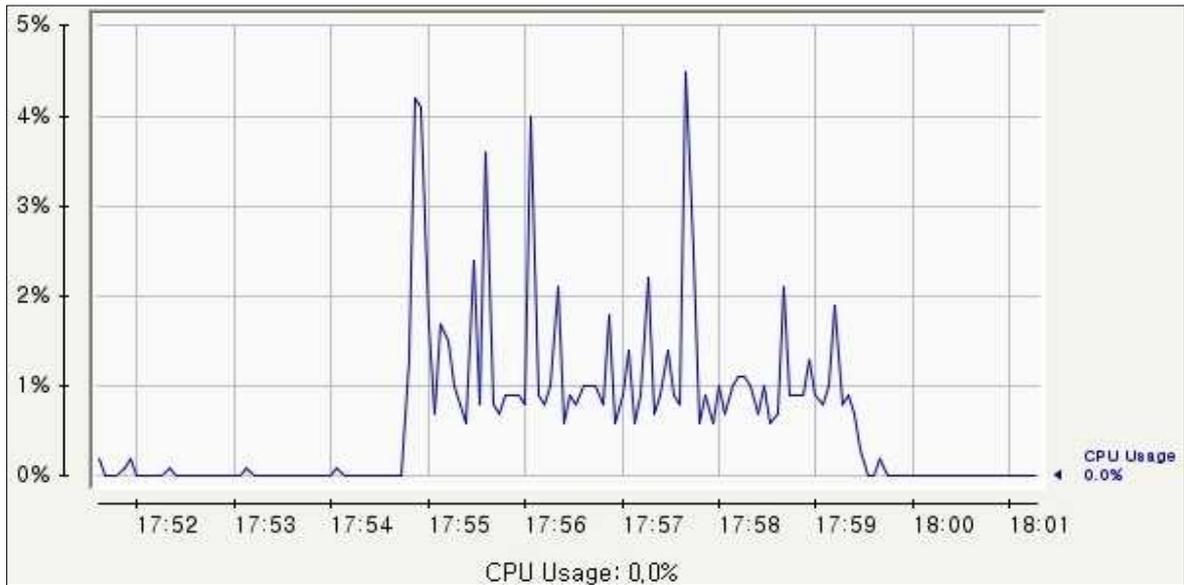


[그림 IV-21. CPU 자원사용률]

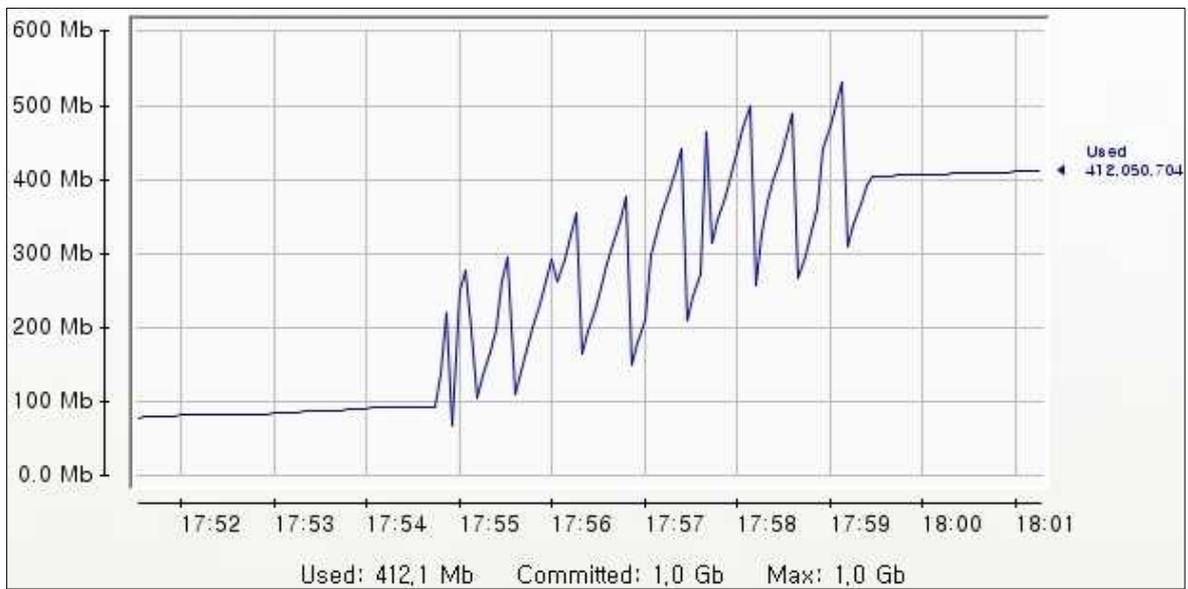


[그림 IV-22. Memory 자원사용률]

- 클러스터 B노드



[그림 IV-23. CPU 자원사용률]



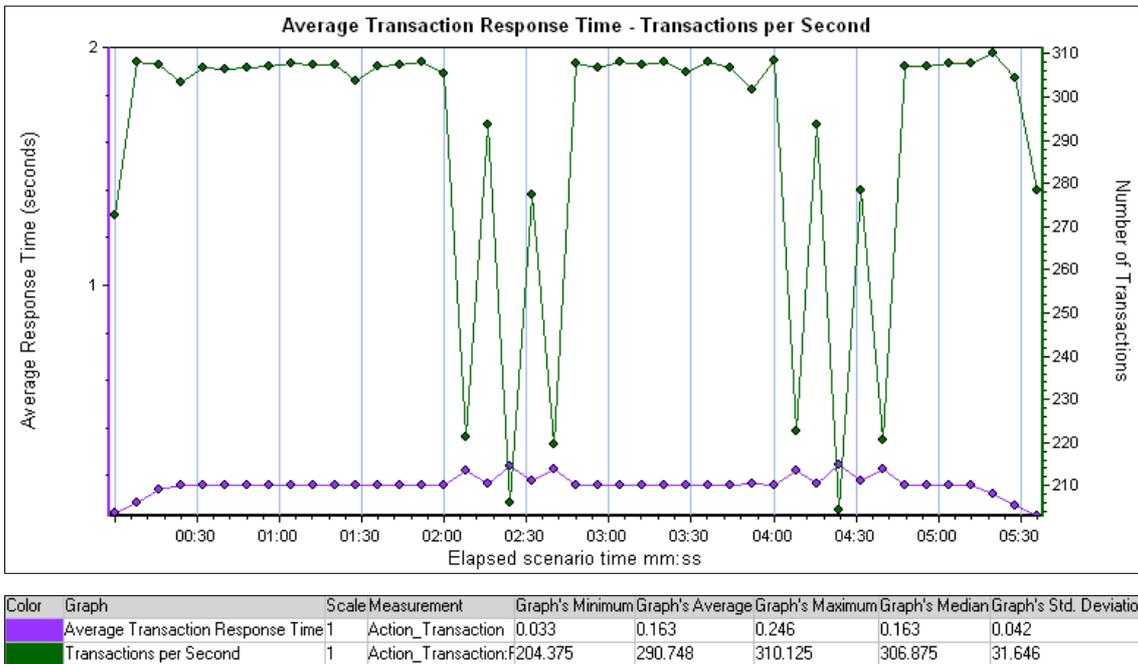
[그림 IV-24. Memory 자원사용률]

□ 클러스터 랜덤 입력

- 수행정보

부하발생	<ul style="list-style-type: none"> - Lamp up : 5User / 1초 - Running-Time : 5분 - Lamp down : 5User / 1초 - 동시사용자 : 50User
수행 조건	- UUID를 생성하여 중복되지 않은 키 생성 및 데이터 입력

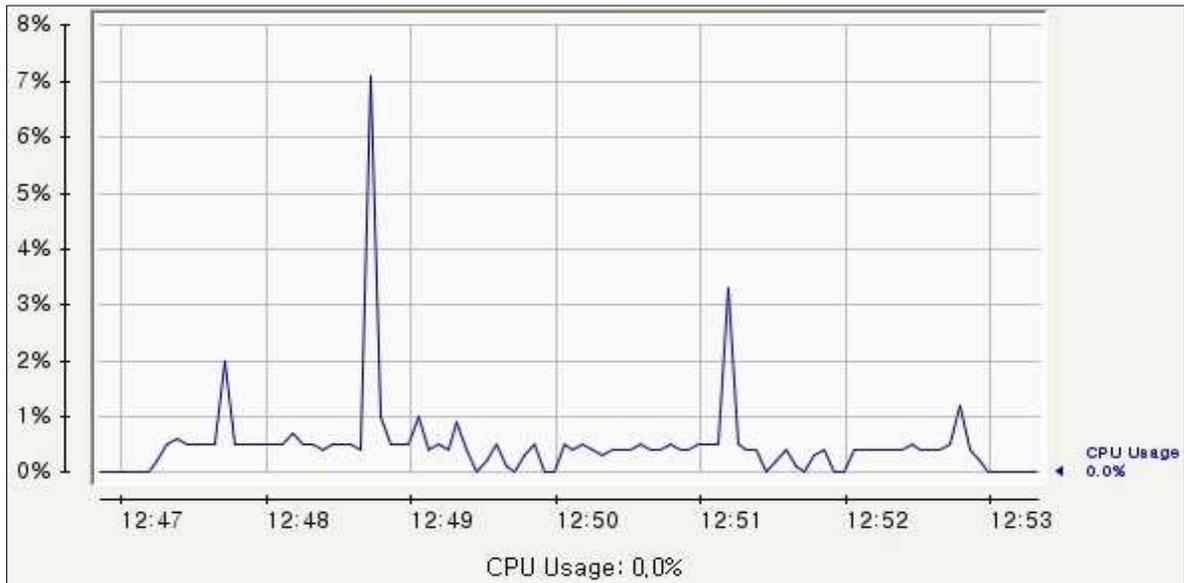
- 측정결과 (A Stack)



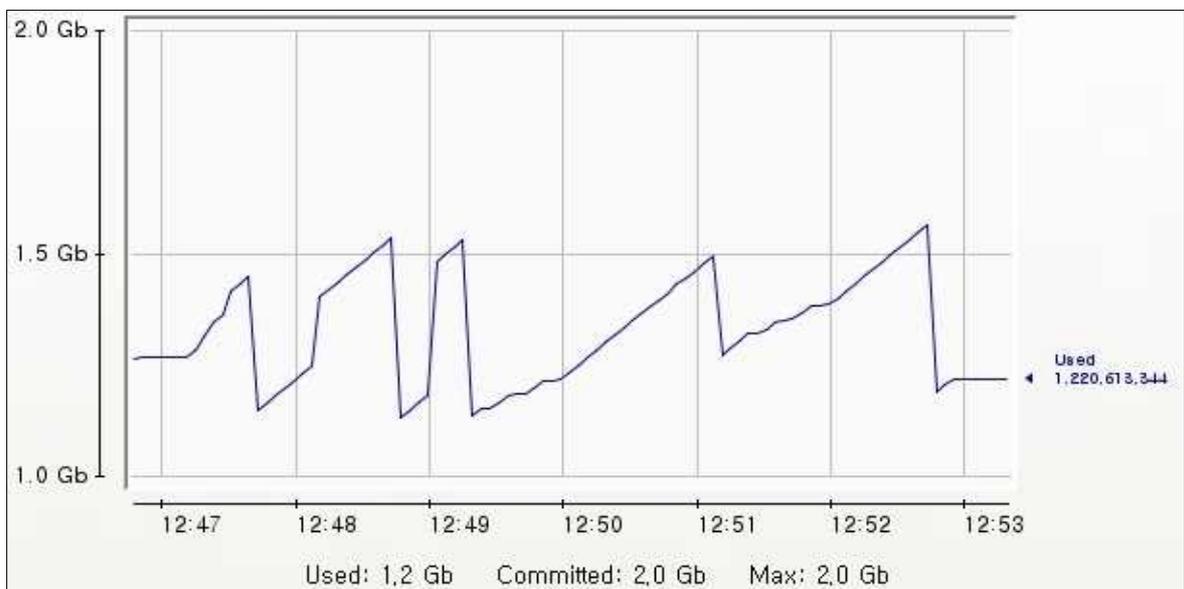
[그림 IV-25. 트랜잭션 결과]

응답시간은 평균 1초 이내의 안정적인 응답을 보이고 있음

- 클러스터 A노드 (Seed서버)

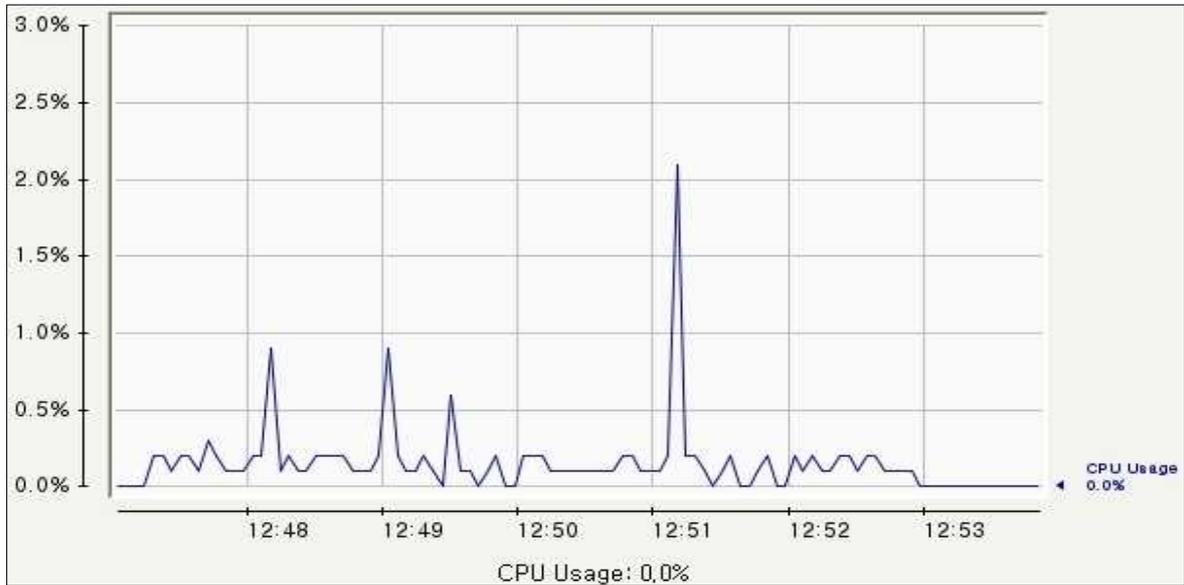


[그림 IV-26. CPU 자원사용률]

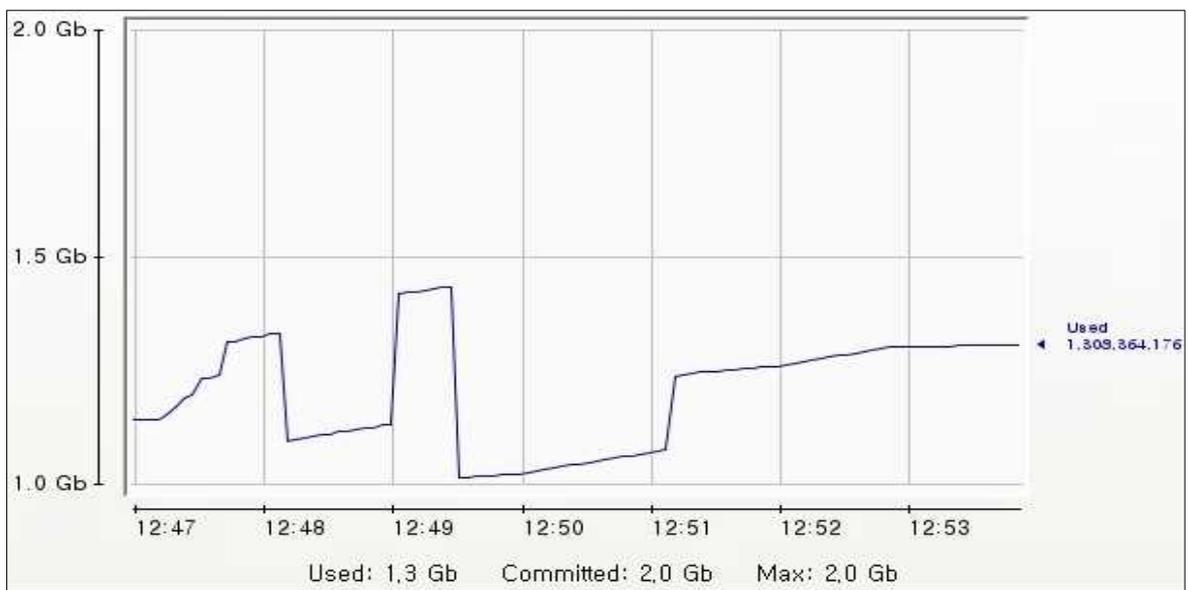


[그림 IV-27. Memory 자원사용률]

- 클러스터 B노드

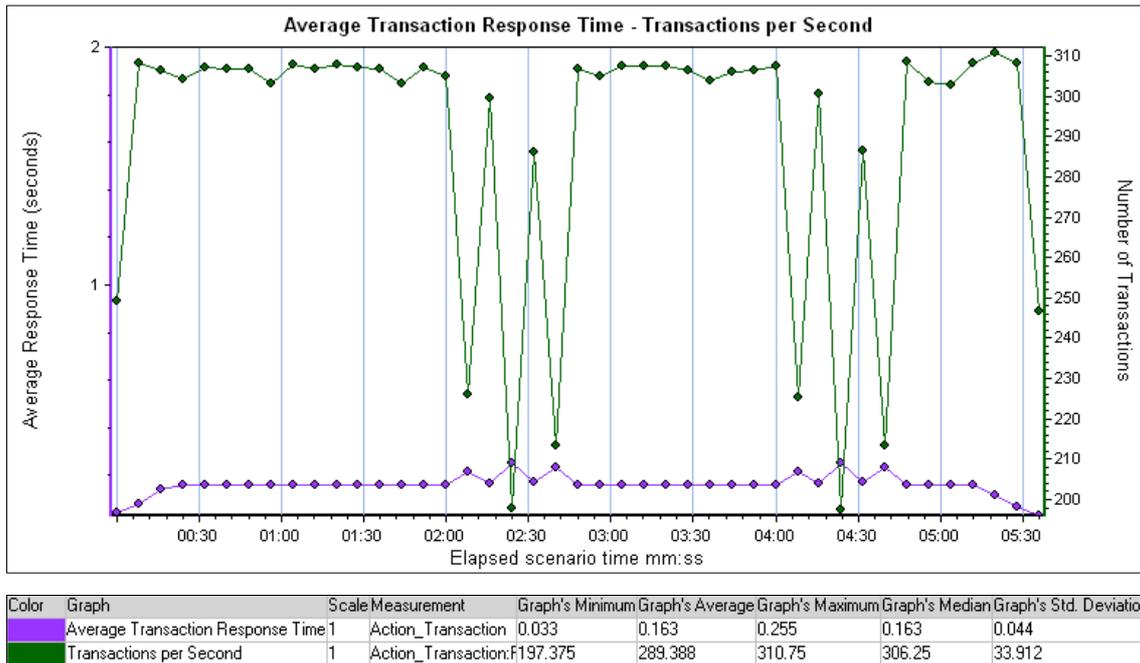


[그림 IV-28. CPU 자원사용률]



[그림 IV-29. Memory 자원사용률]

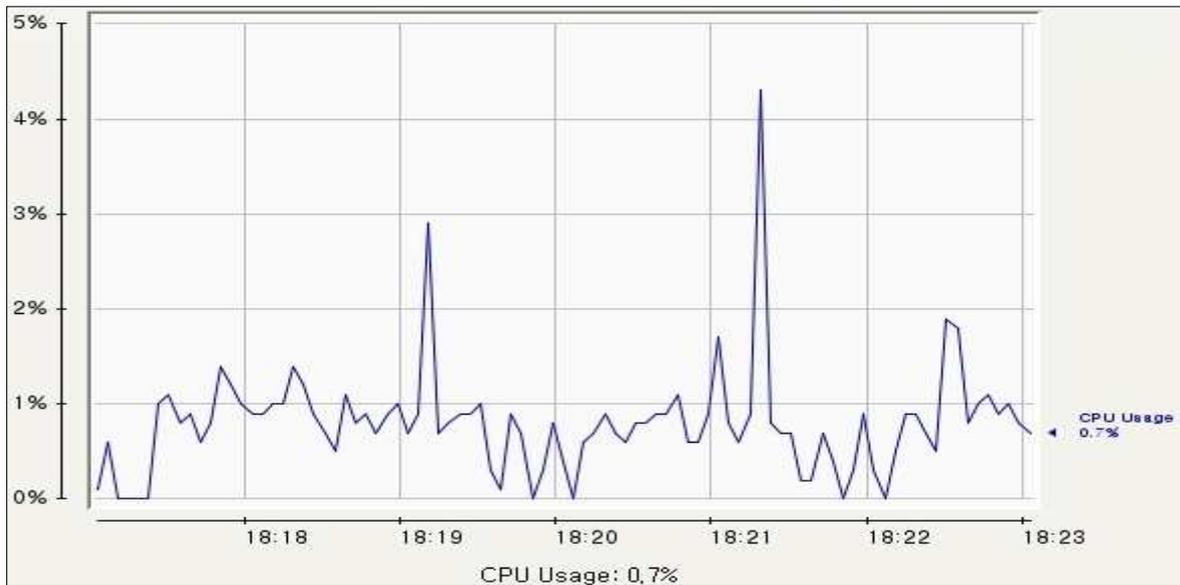
- 측정결과 (B Stack)



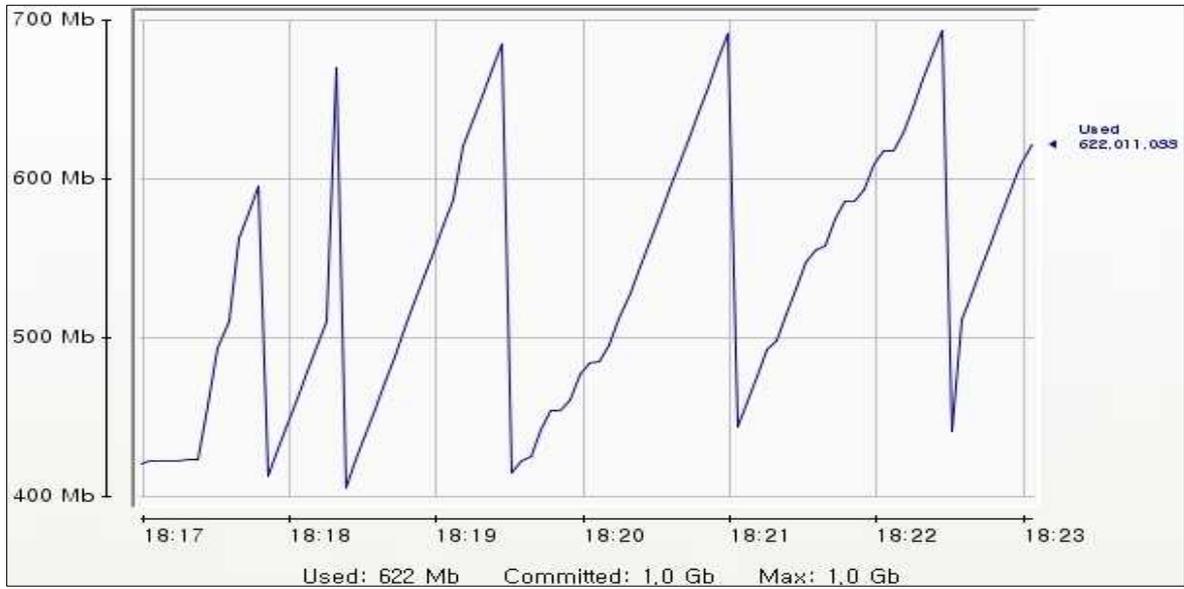
[그림 IV-30. 트랜잭션 결과]

응답시간은 평균 1초 이내의 안정적인 응답을 보이고 있음

- 클러스터 A노드 (Seed서버)

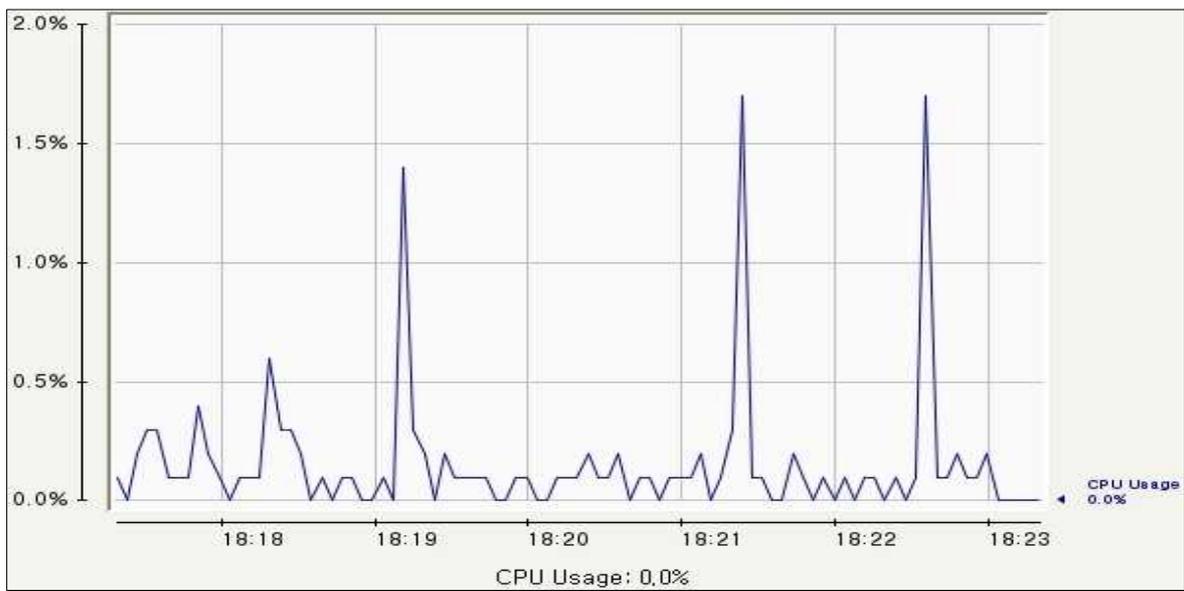


[그림 IV-31. CPU 자원사용률]

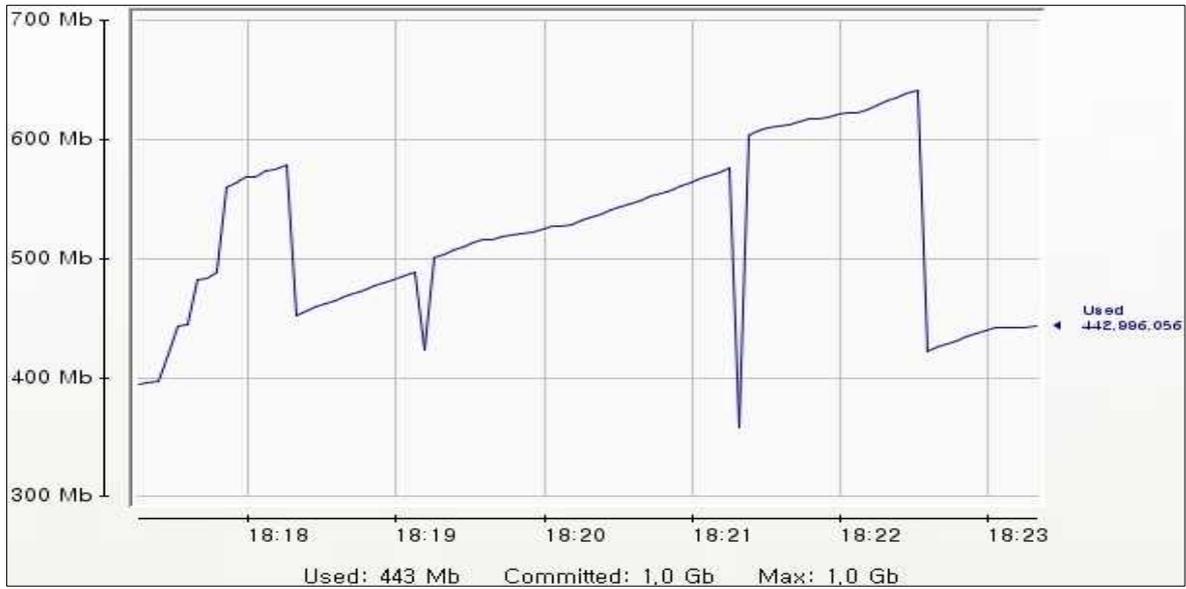


[그림 IV-32. Memory 자원사용률]

- 클러스터 B노드



[그림 IV-33. CPU 자원사용률]



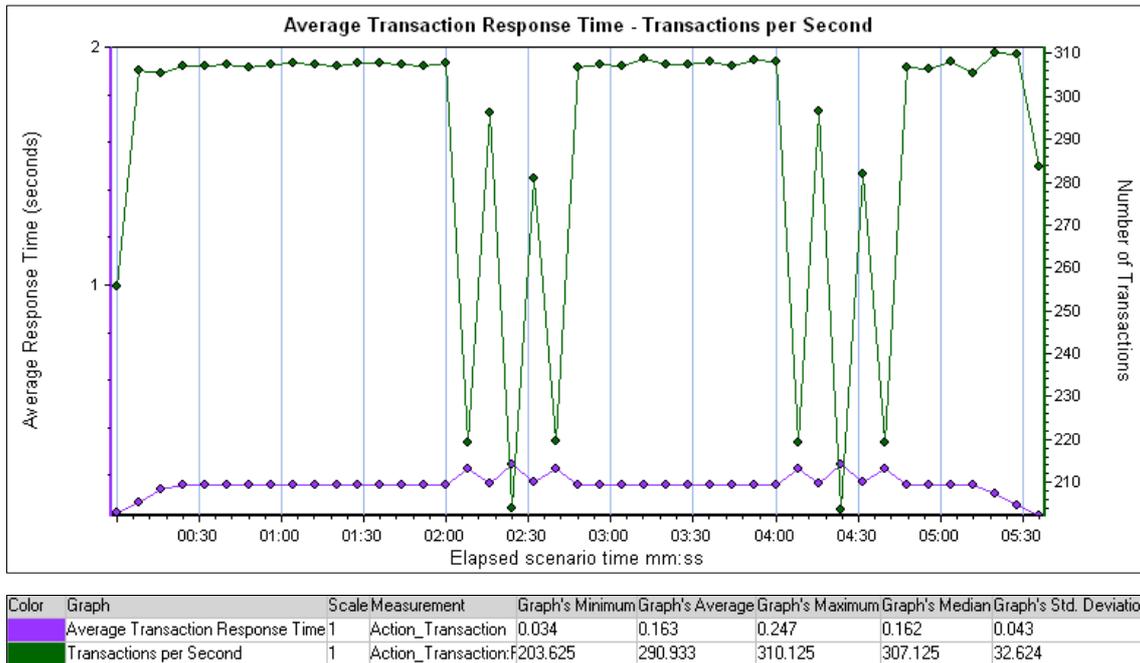
[그림 IV-34. Memory 자원사용률]

□ 클러스터 랜덤 조회

- 수행정보

부하발생	<ul style="list-style-type: none"> - Lamp up : 5User / 1초 - Running-Time : 5분 - Lamp down : 5User / 1초 - 동시사용자 : 50User
수행 조건	<ul style="list-style-type: none"> - CF(Column Family)에 등록 된 키 값을 랜덤하게 생성한 후 데이터 조회 - \$bbs_key = rand(1, 500000);

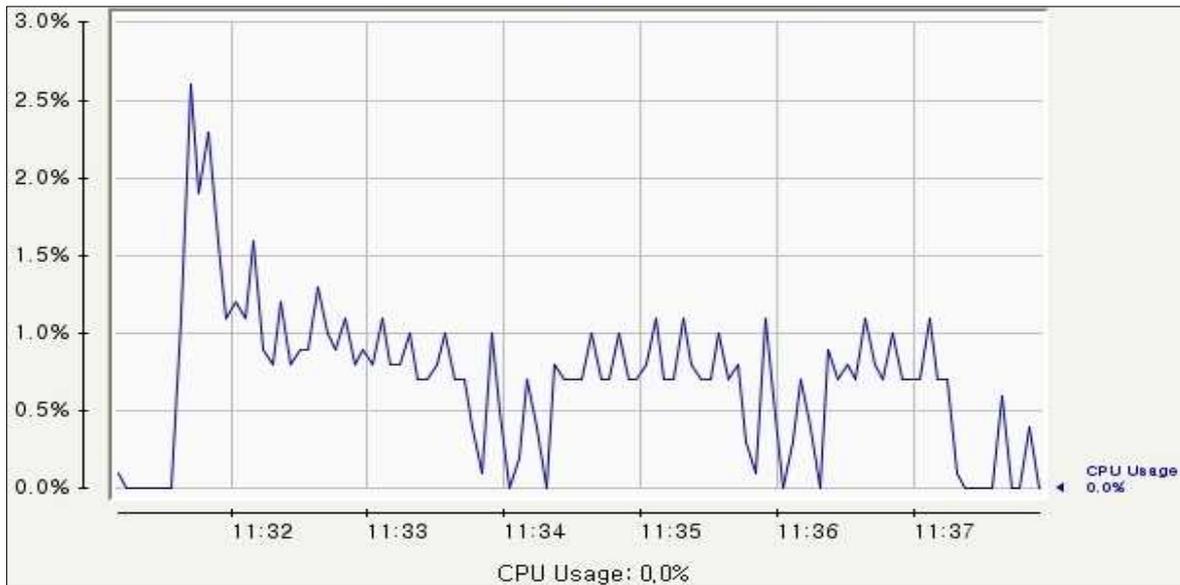
- 측정결과 (A Stack)



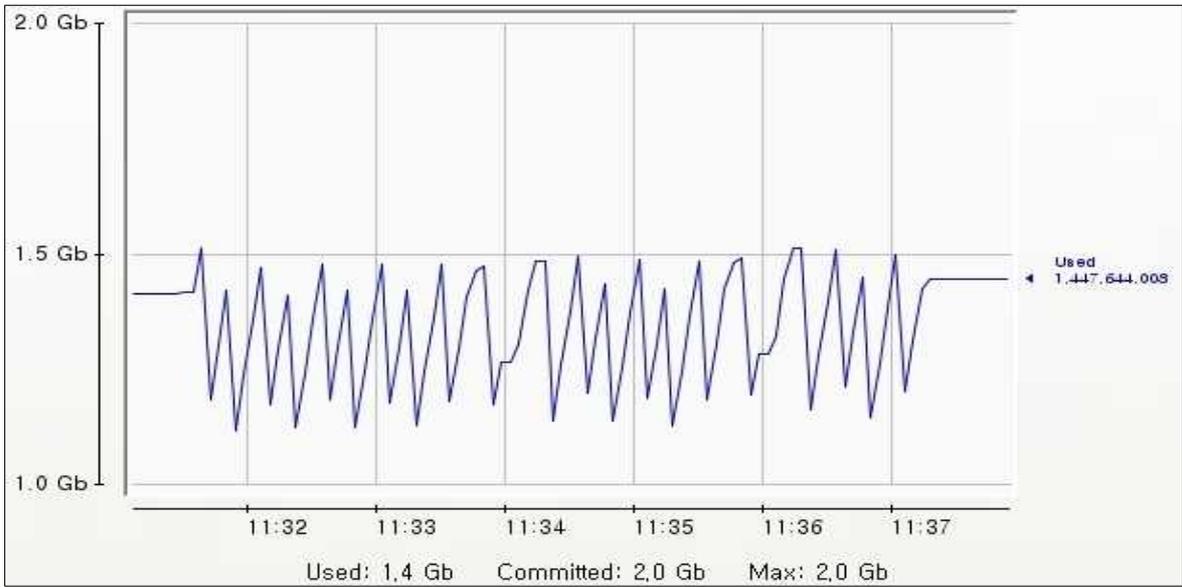
[그림 IV-35. 트랜잭션 결과]

응답시간은 평균 1초 이내의 안정적인 응답을 보이고 있음

- 클러스터 A노드 (Seed서버)

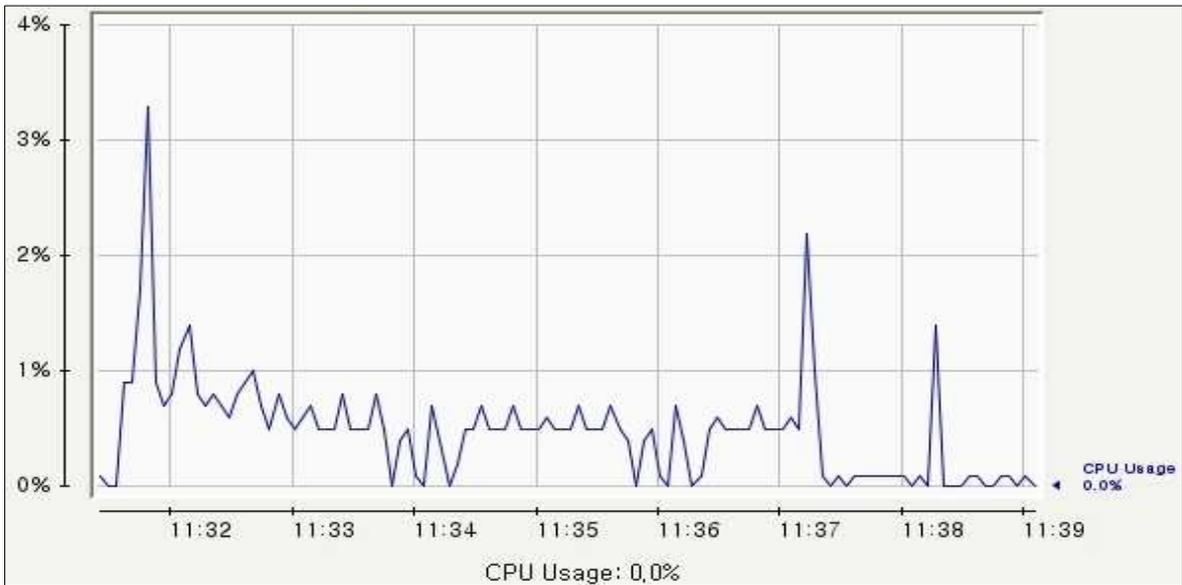


[그림 IV-36. CPU 자원사용률]

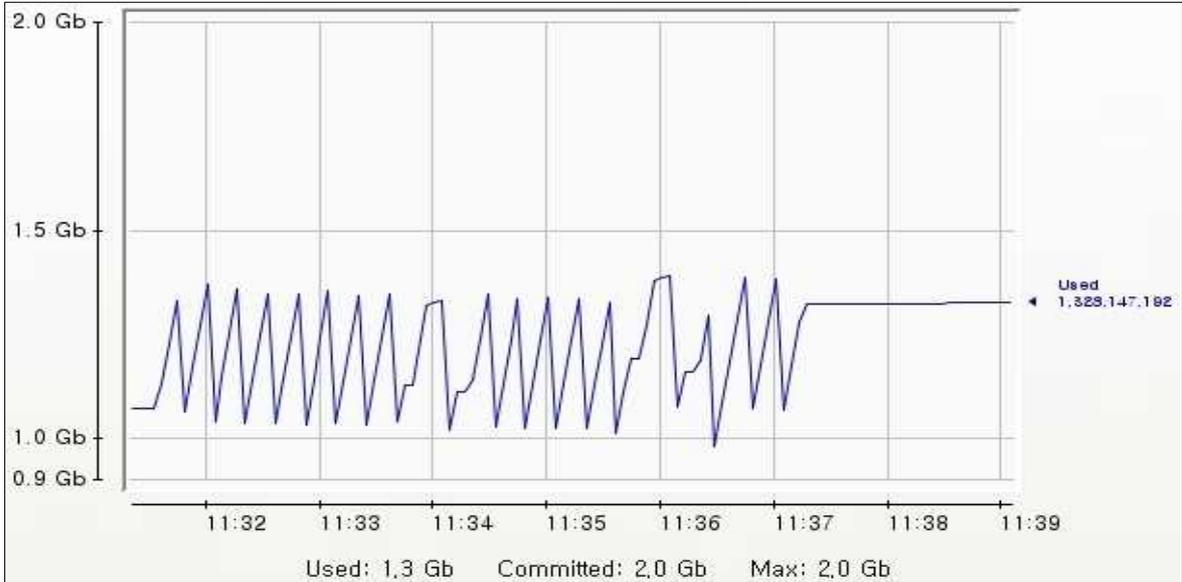


[그림 IV-37. Memory 자원사용률]

- 클러스터 B노드

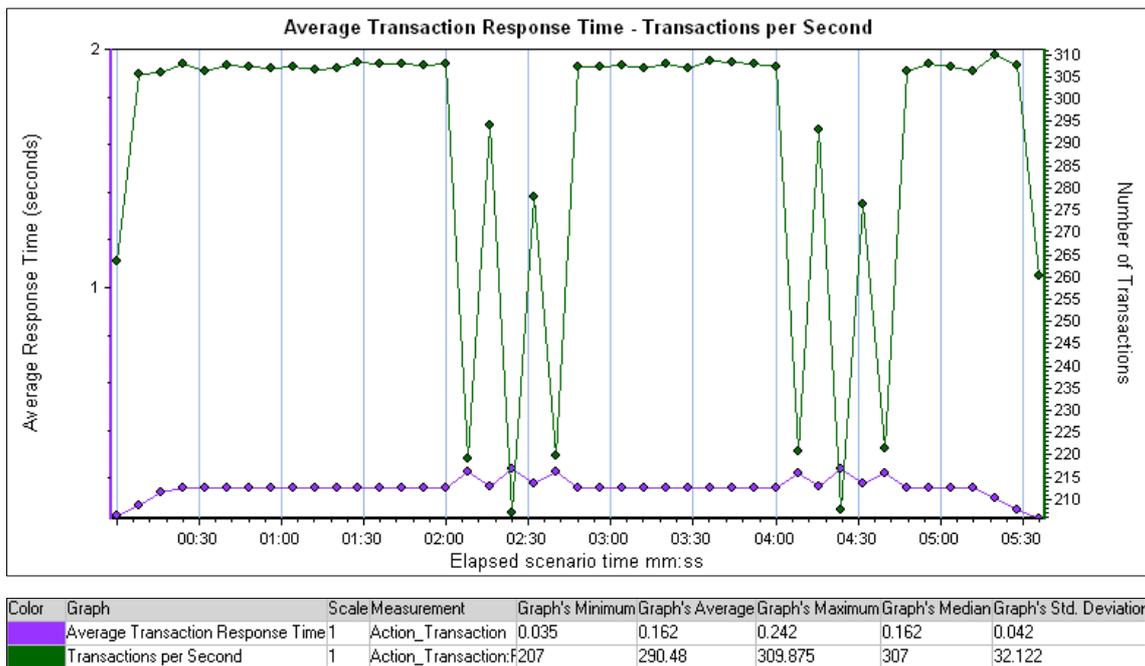


[그림 IV-38. CPU 자원사용률]



[그림 IV-39. Memory 자원사용률]

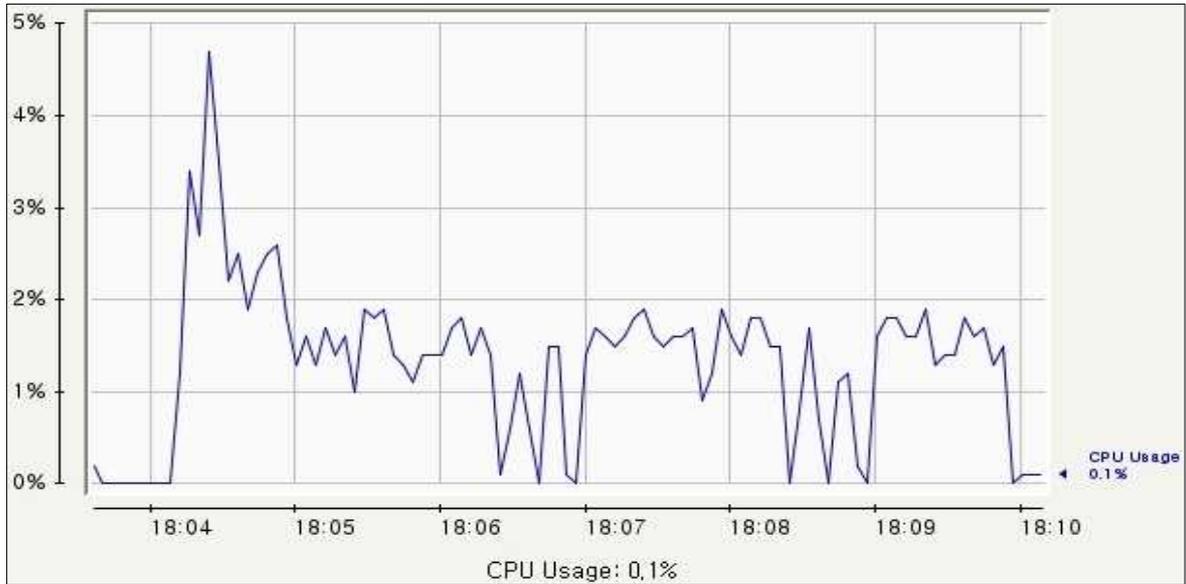
- 측정결과 (A Stack)



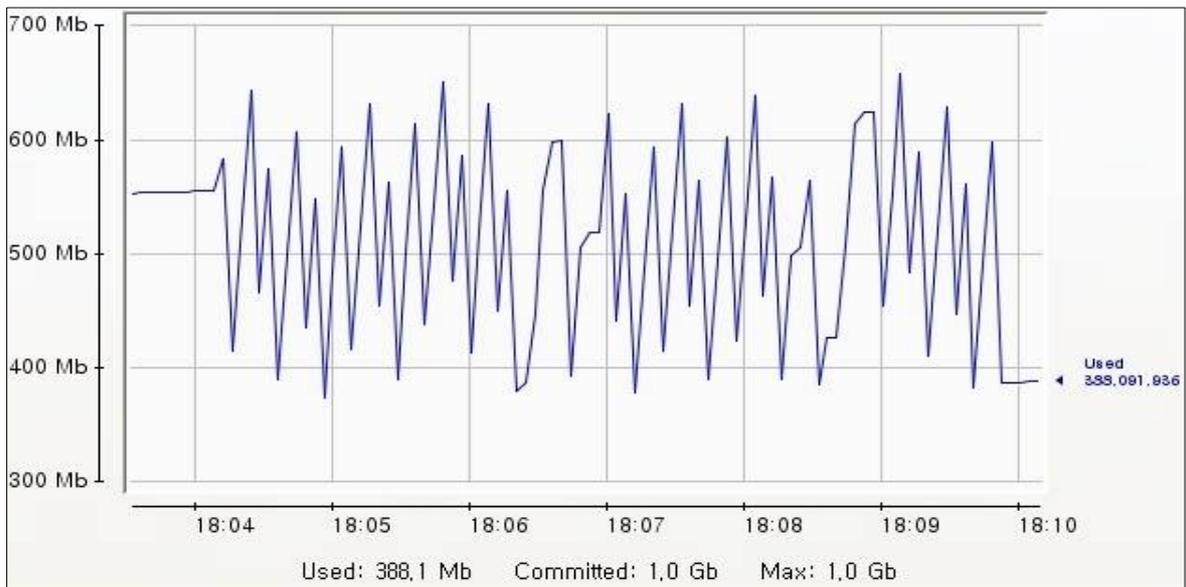
[그림 IV-40. 트랜잭션 결과]

응답시간은 평균 1초 이내의 안정적인 응답을 보이고 있음

- 클러스터 A노드 (Seed서버)

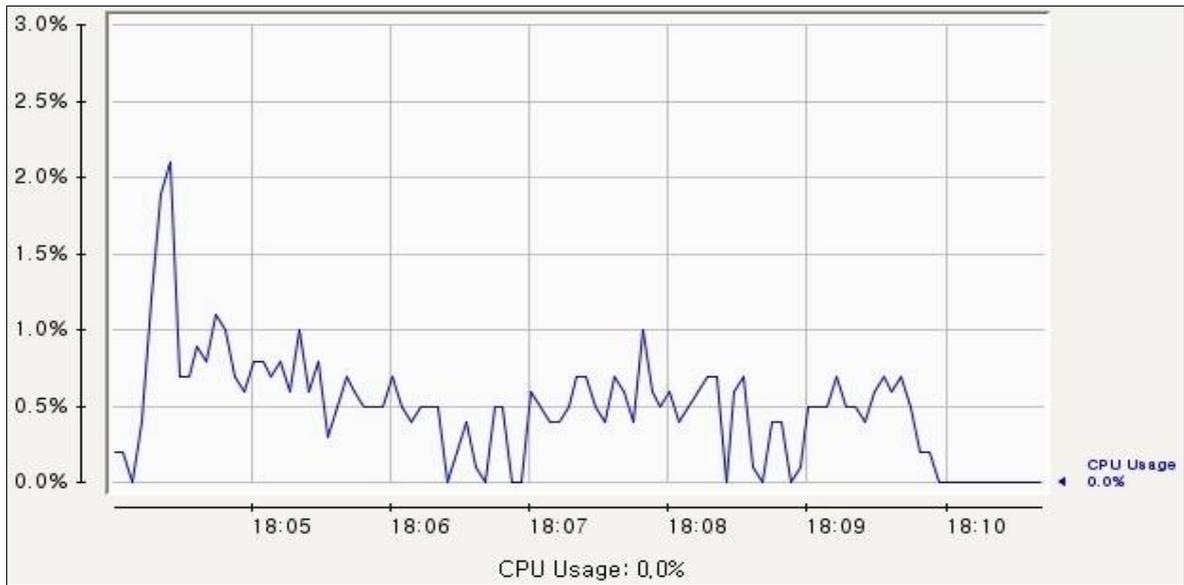


[그림 IV-41. CPU 자원사용률]

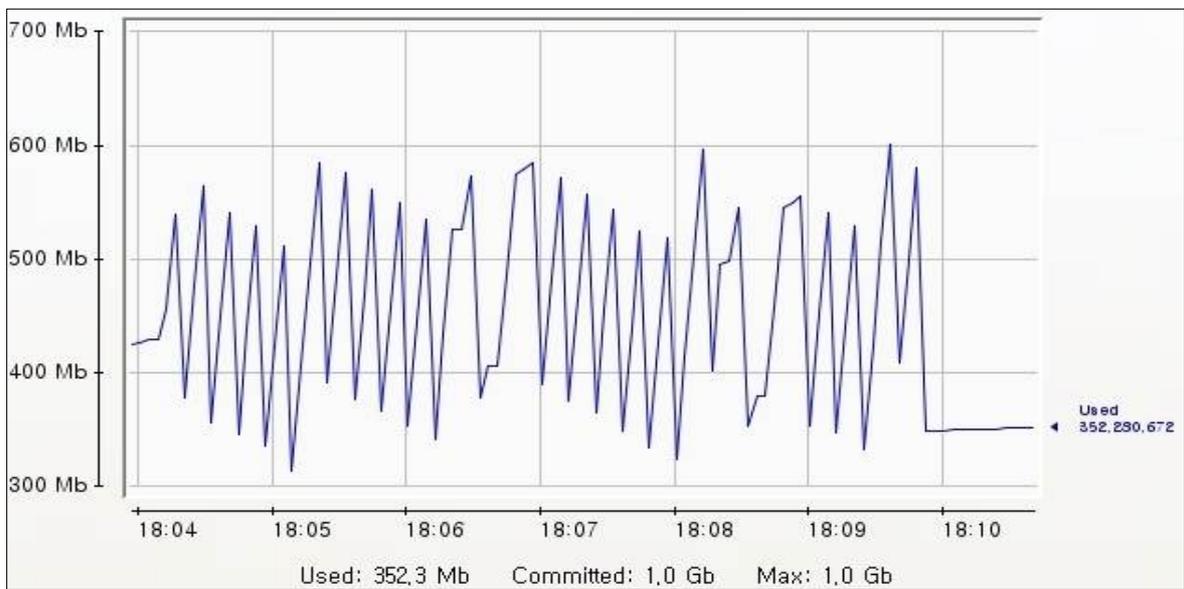


[그림 IV-42. Memory 자원사용률]

- 클러스터 B노드



[그림 IV-43. CPU 자원사용률]



[그림 IV-44. Memory 자원사용률]

V. 종합

- Cassandra Stack 통합테스트 수행 결과 공개SW로 구성된 A, B Stack 상에서 각 기능 시나리오 수행 시 일부 기능에서 결함 (기능 테스트 결과 참조)이 발생하였으며, 구체적인 원인을 확인하기 위한 추가적인 내용 파악이 필요할 것으로 판단됨

- 최근, 디지털 공간에서 소셜 미디어, 영상, 음성 등 비정형 데이터가 크게 늘고 있으며, 이러한 빅데이터에 대한 처리 및 분석에 많은 관심이 증가하면서 Cassandra 분산DB에 대한 관심도 확대되고 있어, 향후 Cassandra를 도입하고자 하는 공공 및 민간에서 본 통합테스트의 Stack 환경 통합테스트 정보가 유용하게 활용되기를 기대함

※ 참고 자료

- [1] <http://cassandra.apache.org>
- [2] <http://code.google.com/p/simpletools-php>
- [3] http://en.wikipedia.org/wiki/Apache_Cassandra
- [4] <http://www.apache.or.kr/>
- [5] <http://www.php.net>
- [6] <http://centos.org>
- [7] <http://www.microsoft.com/windowsserver2008/ko/kr/>
- [8] Cassandra 카산드라 완벽 가이드 - 한빛미디어