

S/W 성능 벤치마크 테스트 계획 수립 : 사례를 중심으로

정도균, 정창신, 신석규
한국정보통신기술협회 시험인증연구소 S/W시험인증센터

The S/W Performance Benchmark Test Planning : Focused on the Cases

Do-Kyun Jeong*, Chang-shin Jeong, Seok-Kyoo Shin
*TTA ITTCL SQEC

E-mail : dkjeong@tta.or.kr, cschung@tta.or.kr, skshin@tta.or.kr

요약

소프트웨어 산업은 급속한 기술의 발전과 적용분야의 복잡성 및 다양성 등으로 제품 품질 및 성능에 대한 정확한 평가분석이 어렵다. 그러나 산업에서 필요로 하는 소프트웨어를 적재적소에 공급하거나 신규 시장 진출을 위한 마케팅 활동, 동종 소프트웨어간의 비교평가를 통한 제품 개선 등과 같은 요구사항의 증대로 소프트웨어 벤치마킹의 중요성이 더욱 부각되고 있는 상황이다. 또한 이러한 제품 간 비교·평가를 올바르게 수행하기 위해서는 신뢰할 만한 결과의 획득과 효율적인 수행이 필수적이다. 이에 본 논문에서는 주로 성능측면의 소프트웨어 벤치마크 테스트를 수행하는데 있어 현실적이고 신뢰할만한 그리고 정확한 사용자 요구사항을 반영할 수 있도록 벤치마크 테스트 계획 수립과 설계의 주요 기본단계를 경험적 지식에 기반을 두어 사례 중심으로 소개하고자 한다.

1. 서론

1.1. 연구의 필요성

최근의 산업은 하드웨어보다는 소프트웨어가 점점 주류를 이루어 가고 있으며 사용자(구매자)에게는 적절하고 객관적으로 우수 제품을 선택할 수 있는 제품의 비교정보가 필요하고 개발업체에게는 타 경쟁제품과의 비교를 통한 장·단점 파악 및 취약점 보완·개선/마케팅 활동 지원 등을 통한 경쟁력 제고방안이 절실하다.

또한 열악한 환경 속에 있는 국내 소프트웨어 개발업체를 선도하고 막연한 외산 소프트웨어 선호

사상을 불식할 수 있는 등의 다양한 서비스 개발이 산업계 측면에서 필요하다.

이러한 시장 요구사항의 확대에 따라 소프트웨어 벤치마크 테스트의 필요성이 점차적으로 증가하고 그에 따른 표준화된 소프트웨어 벤치마크 테스트 절차와 평가기준의 개발이 절실하다.

선진 외국의 경우에는 오래전부터 소프트웨어 벤치마크 테스트의 중요성을 인식하고 관련분야의 연구 및 활동을 공공기관과 업체를 중심으로 활발히 추진하고 있으며 국내의 경우 한국정보통신기술협회(TTA) S/W시험인증센터와 한국소프트웨어진흥원(KIPA)에서 국내 소프트웨어 벤치마크 테스

트 활성화를 위한 노력을 기울이고 있다. 그러나 국내 전문 벤치마크 테스트 업체가 전무하고 개발 업체와 소비자의 활용정도가 높지 않은 것이 현실이다.

이는 정확한 수치가 나오는 하드웨어 및 네트워크 제품과는 달리 소프트웨어는 객관적인 평가가 어렵다는 점에서도 그 요인을 찾을 수 있다.

따라서 소프트웨어 개발업체, 소비자, 평가자 등 의뢰자에 따라 다른 결과가 발생하는 경우가 종종 발생하고 그 결과의 신뢰성 및 객관성 확보가 중요한 이슈로 부각되고 있는 상태에서 소프트웨어 벤치마크 테스트가 수행주체에 상관없이 객관적인 입장에서 수행되고 신뢰할 만한 결과를 도출할 방안이 조속히 마련되어야 한다.

1.2. 연구의 범위 및 방법

본 논문에서는 일반적인 소프트웨어 품질 평가 프로세스 및 소프트웨어 성능 테스트 기술을 검토·분석하여 성능측면의 소프트웨어 벤치마크 테스트 계획을 수립하는데 있어서의 주요 항목과 지침을 제시하고자 한다.

이러한 주요 항목 및 지침은 서버 OS의 성능 벤치마크 테스트의 일부 사례를 중심으로 경험적 지식에 기반을 두어 제시되며 테스트의 효율적 수행과 객관적이고 신뢰할만한 결과 도출의 기반을 마련하고자 하는데 초점을 둔다.

1.3. 논문구성

본 논문의 구성은 다음과 같다

2장에서는 본 논문에서 제시하는 지침과 관련된 기술을 소개한다. 일반적으로 소프트웨어의 품질평가에 사용되어지는 프로세스에 대해서 알아보고 구체적으로 사용되어지는 소프트웨어 성능 테스트 기술에 대해서 소개한다.

3장에서는 경험적 지식에 기반으로 하여 효율적으

로 테스트를 수행하고 결과에 대한 객관성, 신뢰성을 초기(테스트 계획 및 설계)단계에서 확보하는데 필요한 주요 항목과 관련 지침을 제시한다.

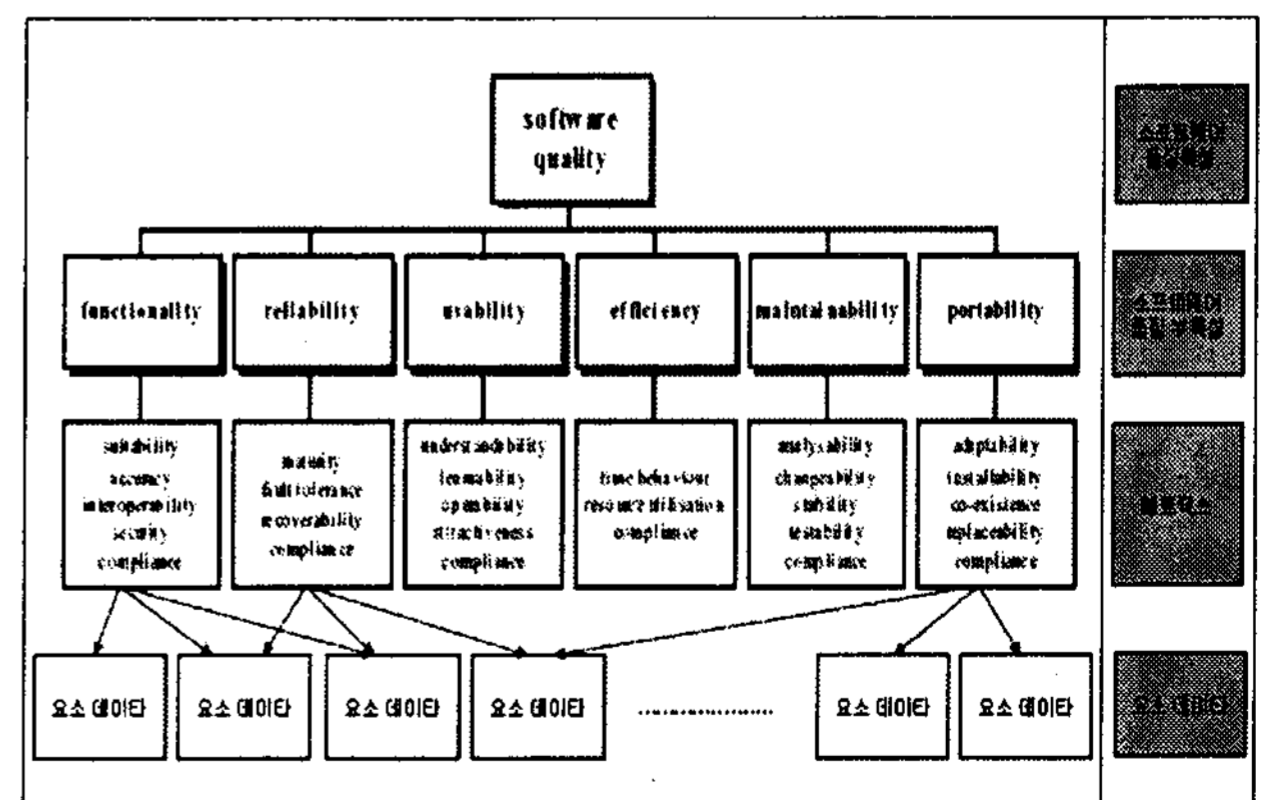
이어서, 4장에서는 제시 항목에 대한 신뢰성 확보 차원에서 상용 서버 OS를 대상으로 한 성능 벤치마크 테스트의 일부 사례연구를 분석한다.

마지막으로 5장에서는 결론 및 향후 연구방향에 대해 기술한다.

2. 관련연구

2.1. 소프트웨어 품질 평가 프로세스

IT산업에서의 소프트웨어 중요성 및 가치가 하드웨어보다 점차 가중됨에 따라 소프트웨어의 품질에 대한 중요성 또한 증대되어 왔다. 이러한 요구 사항에 부응하기 위한 소프트웨어의 품질을 관리하기 위해서는 소프트웨어 제품의 품질(개발 과정의 품질을 포함)을 객관적이고 정량적으로 규정·평가하는 기술이 가장 중요하다. ISO/IEC 9126 표준은 이러한 소프트웨어 품질을 평가하기 위한 기본 프레임워크를 제공하는 문서의 일부로 작성되었다.

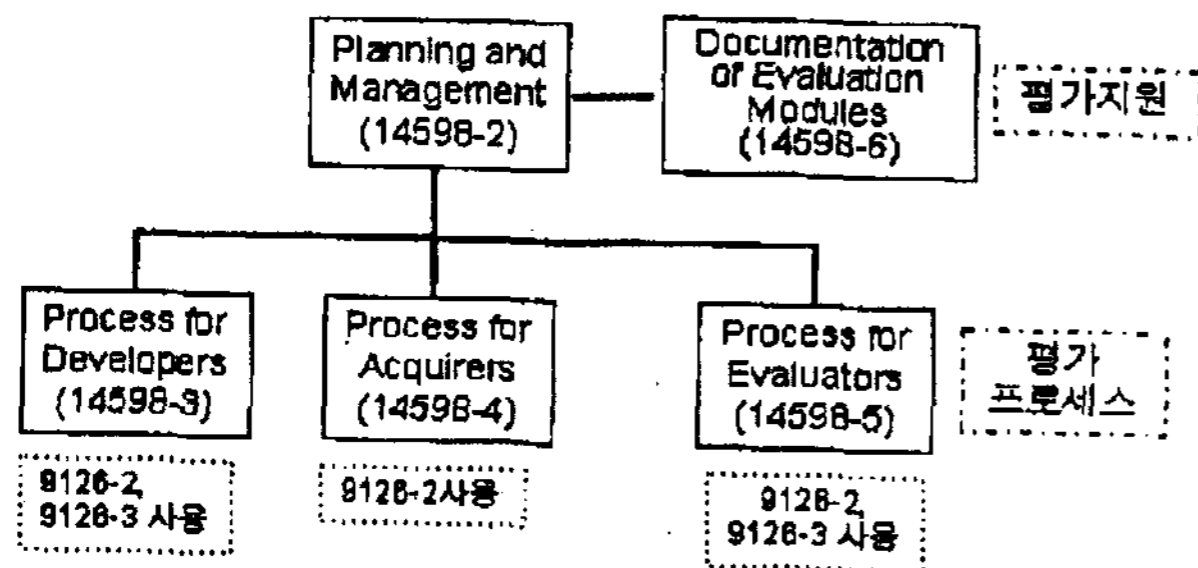


[그림 1] 소프트웨어 품질 특성 및 평가 접근방법

[그림 1]은 ISO/IEC 9126시리즈에서 제시하는 품질 모형으로 각 소프트웨어의 품질특성과 이를 이루는 부특성, 각 부특성을 측정하기 위한 메트릭과 이를 구성하는 요소데이터로 구분되어진다. 이 같

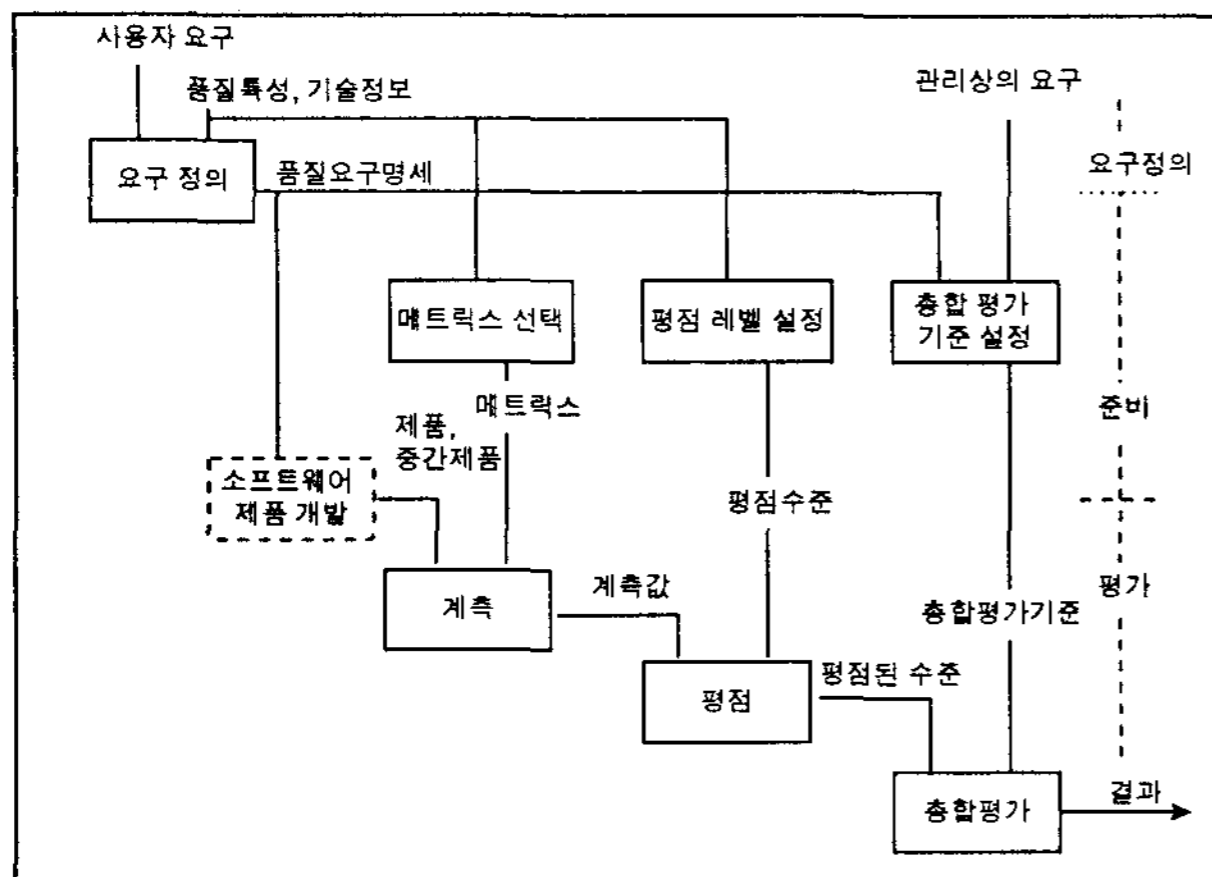
은 소프트웨어 품질 평가방법은 개발자, 획득자, 평가자의 관점에서의 제품 평가 절차 및 지원기술을 제시하고 있는 ISO/IEC 14598 시리즈와 함께 소프트웨어의 품질 특성을 세분화하여 실질적으로 평가하기 위한 프레임워크로 사용된다.

다음 [그림 2]는 품질 평가를 필요로 하는 3가지의 이해 집단(개발자, 획득자, 평가자) 평가 절차 기술을 기술하고 있는 ISO/IEC 14598 시리즈가 ISO/IEC 9126 규정한 품질 모형 표준을 따르고 있음을 보인다.



[그림 2] ISO/IEC 14598 시리즈 구조 및 ISO/IEC 9126 척도표준

이러한 품질 평가의 프로세스는 다음 [그림 3] '일반적인 소프트웨어의 품질 평가 프로세스'와 같이 크게 요구단계, 준비단계 및 평가단계의 3단계로 분류되어질 수 있다.



[그림 3] 일반적인 소프트웨어 품질 평가 프로세스

요구단계는 품질요구를 명확히 하는 단계이고 준비단계는 평가를 행하기 위한 단계로 매트릭 선택, 평점 수준 설정, 총합평가의 기준 등을 설정한다.

평가 단계는 실제로 평가를 수행하는 단계로 측정, 평점, 평가가 있으며 본 논문에서 제시하고자 하는 범위는 [그림 3]의 요구단계·준비단계에 해당한다.

2.2. 소프트웨어 성능 테스트 기술

성능 테스트는 일반적으로 소프트웨어에 대한 시간 프로파일링, 실행흐름, 응답시간, 운용상의 신뢰성, 제약사항 등과 같은 테스트 대상의 성능관련 특성을 평가하기 위해 구현·실행된다.

초기 아키텍처 반복단계에서는 아키텍처와 관련된 성능 병목지점(Bottleneck)을 식별하고 제거하는데 주목하며 구축단계에서는 소프트웨어 응답시간과 자원 등의 환경을 최적화하고 다수의 트랜잭션, 클라이언트 또는 데이터의 양과 같이 소프트웨어와 복합적으로 구성되어 있는 시스템(소프트웨어, 하드웨어, 네트워크)이 높은 부하레벨 및 극한환경을 다룰 수 있다는 것을 검증하기 위하여 구현되어지고 설계되어진다.

이러한 성능 테스트는 운영환경(Operating Environment)에서의 시스템 성능을 평가하기 위하여 실제 사용자의 행위를 현실 세계를 반영할 수 있도록 시뮬레이션 함으로써 전자 상거래(E-Commerce), 클라이언트-서버(Client-Server)와 같은 다중 사용자 분산 시스템(Multi-User Distributed System)을 훈련시키는 프로세스이다.

따라서 실제 운영환경에서 발생하거나, 발생할 수 있는 현실적인 부하레벨의 설정과 생성이 매우 중요하며 이에 대한 일반적인 성능 테스트 방법은 크게 모델링 방법과 경험적인 방법이 있다.

다음 [표 1]은 일반적인 성능 테스트 방법에 대해서 기술한다.

[표 1] 일반적인 성능 테스트 방법

구분	특징
모델링 방법	<ul style="list-style-type: none"> · 테스트 대상 시스템을 필요로 하지 않음 · 모델에 의해 투영된 성능을 측정 · 입력물의 유용성에 의존하여 결과를 산출
경험적인 방법	<ul style="list-style-type: none"> · 테스트 대상 시스템이 반드시 필요 · 테스트 대상 시스템에 대해 직접적으로 측정 · 실제 부하 또는 테스트 부하를 사용할 수 있음 · 테스트 환경에 의존하여 결과가 산출됨

모델링 방법은 역사적인 데이터(Historical Data)나 데이터베이스, 하드웨어, 소프트웨어와 네트워크 같은 입력물을 사용하는 통계적인 모델링 소프트웨어를 사용한다. 이러한 모델링 기법은 알려진 시스템의 특성을 기초로 하여 투영된 성능의 측정치를 계산한다.

경험적인 방법은 실제 소프트웨어 운영환경을 이용한다. 따라서 이 방법은 직접적인 측정을 수행할 기능적인 테스트 대상 시스템을 필요로 하고 실제 부하를 이용하거나 테스트 부하를 이용하여 수행하게 된다. 이러한 접근방법은 모델링 방법에 비해 매우 현실적이고 테스트 도구를 포함하여 테스트 대상 시스템에 대한 충분한 결과의 획득이 가능함으로 소프트웨어에 대한 성능 벤치마크 테스트에 매우 적합하다.

3. 본론

3.1. 개요

본 논문에서 제시하고자 하는 계획 수립의 주요 항목 및 지침의 범위는 [그림 3] 일반적인 소프트웨어 품질 평가 프로세스의 요구·준비단계에 해당하며 최종적으로는 객관적이며 명확하고 간결한 소프트웨어 성능 벤치마크 테스트 계획 수립, 문서화하는데 있다.

즉, 소프트웨어 벤치마크 테스트 계획 수립이란 “체계적이고 객관적으로 소프트웨어를 측정·시험·평가하기 위한 항목들의 집합과 명세”를 의미하며

구체적으로는 테스트 목적, 항목, 측정 항목 및 평가방법 등을 사전적으로 정의하며 문서화 시 결과에 대한 객관성 및 신뢰성 확보 방안을 함께 마련하여 제시하는 것이다.

이러한 접근은 일반적인 소프트웨어 개발 프로세스나 방법론에서 제시하는 것과 유사하게 수행 초기 단계에서부터 명확하게 (테스트)요구사항을 정의하고 발생 가능한 위험요소를 최소화하여 궁극적으로 성공적인 프로젝트를 달성하는데 있으며 경험적 지식에 근거하여 작성되었다.

본 논문에서는 다음 [표 2]와 같이 크게 3가지로 구분하여 필요한 활동들과 지침들을 기술하며 각 단계들은 점증적이고 반복적이며 병렬적으로 수행되어야 한다.

[표 2] S/W 성능 벤치마크 테스트 계획 수립

단계	주요 활동
요구사항 식별 및 정제	<ul style="list-style-type: none"> · 초기 요구사항 정의 · 옵션 조사 및 추가의견 반영 · 주요 요구사항의 식별 및 정제 · 최종 요구사항의 평가 및 문서화
부하레벨 계획단계	<ul style="list-style-type: none"> · 입력부하 및 본질 이해단계 · 목표 부하레벨 예측단계
계획 문서화	<ul style="list-style-type: none"> · 문서 템플릿 작성 · 문서화 및 정제

3.2. S/W 성능 벤치마크 테스트 계획 단계

3.2.1. 요구사항 식별 및 정제 활동

이 단계에서는 초기 요구사항에 대한 목록을 생성하는 것이 가장 중요하다. 벤치마크 테스트 대상 제품은 어떤 기능을 가지고 있어야 하는지와 운영 환경에서의 효율적 사용을 위해 필요한 능력은 무엇인지에 대한 초기 요구사항을 명세하는 것이 테스트 계획 수립의 시작점이다.

다음은 요구사항 식별 시 고려해야 할 내용을 나타내고 있다.

- 비즈니스 요구사항

- 관련 주요 이슈사항 식별
- 추가 요구사항(옵션) 식별 및 정제
- 주요 요구사항에 대한 우선순위 매김

이 단계에서는 대상 제품 관련자(개발자, 사용자, 구매자, 관련 전문가 등 제 3자)를 참여시킬 수 있으며 너무 많거나 적지 않다면 보유 숙련정도 (Skill Level)가 다를지라도 초기 요구사항을 작성하는데 충분한 역할을 수행할 수 있다.

이러한 다양한 레벨의 사람들을 포함하는 것은 추가적인 요구사항을 밝히는데 유용하다.

만약 다른 수단(뉴스그룹, 웹 검색, 잡지 또는 컨퍼런스 등)을 통해 충분한 관련정보를 수집할 수 있고 시간적 여유가 적거나 참여자와 면밀한 작업을 수행할 수 없다면 배제할 수 있다.

본 활동단계에서의 산출목록은 크게 테스트 목적, 배경, 범위 및 항목 등으로 구성되며 목적에는 벤치마크 테스트를 통해 달성하려는 것이 무엇인지 (제품 간 우선순위 매김, 장·단점 파악, 이전 버전과의 비교 등)와 그에 기반을 두어 무엇이(테스트 범위 및 항목) 수행되어야 하는지 명시적으로 계획서에 기재한다. 또한 참여자들의 요구사항이 충분히 반영되고 최종 합의를 통해 이견이 없어야 한다.

계획 문서의 해당 부분은 대부분 상급자 또는 테스트 인력의 프로젝트 이해를 돕는 것이며 최대한 간결하고 명확하게 평가자의 중립된 입장에서 기재한다.

3.3. S/W 성능 벤치마크 테스트 계획 단계

이 단계는 성능 벤치마크 테스트 시 사용되어질 수 있는 '입력부하 및 본질을 이해하는 활동'과 '목표 부하레벨을 예측하는 활동'으로 크게 구분할 수 있다.

3.3.1. 입력부하 및 본질(특성) 이해단계

테스트 시 구현되어야 하는 성능부하에 대한 본질을 가능한 정확하고 객관적으로 이해하는 단계이므로 '입력부하'와 '사용자 프로파일'을 기반으로 하여 식별을 수행한다.

이러한 입력되는 성능 부하나 사용자 세션을 이해하기 위해서 정의된 요구사항에 따라 시뮬레이션 하거나 실 운영상태에서의 실제 로그파일 또는 로그 분석기 등의 통계 도구 등을 분석에 사용할 수 있다.

사용자 프로파일은 해당 소프트웨어 이용목적을 달성하는데 필요한 사용자 행위관련 변수((예) 메일서버의 경우 사용자 로그인, 메일함 검색, 읽기, 삭제, 쓰기, 첨부 행위 등의 복합 활동)로 반영되어 추후 설계단계에서 성능 테스트 스크립트로 구현되고 입력부하 또는 스트레스 레벨((예) 메일서버의 경우 동시 접속자/메일수신자 수, 첨부파일/메일본문의 크기, 세션의 유지시간 등의 복합)로 반영된다.

[표 3] 입력부하 및 특성 분석

입력모델	테스트 설계 시 반영항목
'사용자 프로파일' 기반	· 사용자 행위 관련 변수로 반영
'입력부하' 기반	· 입력부하 또는 스트레스 레벨로 반영

또한 사용자의 행위는 부하레벨과 결과에 영향을 크게 미치기 때문에 시뮬레이션 또는 로그 등을 통한 사례 분석 시 세심하게 수행되어야 한다.

다음 단계로는 앞서 식별된 부하의 패턴을 이용, 테스트 베드에 가중되는 부하레벨을 결정하는 것이다.

3.3.2. 목표 부하레벨 예측단계

성능 벤치마크 테스트 시 시뮬레이션 되어지는 목표 부하레벨을 예측·결정하는 주요 변수는 다음과 같다.

- '어느 정도까지 성능부하가 증대될 것이라고

미리 예상되는 경우' : 기록 데이터 (Historical Data)가 존재하거나 특정인(조직)에게 질의를 통하여 부하의 목표범위를 예측할 수 있는 경우로 특정 부하레벨에서의 우수 소프트웨어 선정 또는 개선 등을 목적으로 할 경우 주로 사용되며 테스트 범위와 자원 효율성이 높다.

- '발생 가능 트래픽 내에서 처리 가능한 한계 부하레벨을 파악하고자 하는 경우' : 보통 테스트 베드 상에서 구현 가능한 한계 레벨까지 그 부하가 확장되며 전반적인 범위에서의 성능 강·약점을 분석하고자 할 경우 또는 성능병목 원인을 진단하려고 하는 경우 주로 사용된다. 단 테스트 범위가 넓고 자원이 소모가 많다는 약점을 가진다.
- '얼마나 빨리 한계 부하레벨까지 도달하는지' 또는 '한계 로드 레벨에서 얼마나 오랫동안 운영되는지 판단하고자 하는 경우' : 부하레벨을 특정시간 동안 유지하거나 Ramp-Up/Ramp-Down하여 주요 시스템 컴포넌트(하드웨어, 소프트웨어, 네트워크 등)간의 구성에서 시스템 가용성·신뢰성을 파악하거나 테스트 시나리오를 복합 구성하여 시험을 수행하고자 할 때 주로 사용된다. (Ramp-Up/Ramp-Down의 주기는 테스트 베드 자원을 초기화 하거나 테스트 결과간의 부작용(Side-Effect)을 최소화 등의 특정 목적을 위해 설정되어 질 수 있다.)

이러한 변수는 특정한 테스트 목적을 달성하거나 테스트 자원의 효율적 사용을 위해 복합적으로 구성되어 설계에 반영할 수 있다. 또한 소프트웨어는 하드웨어, 소프트웨어, 네트워크 컴포넌트간의 복잡한 시스템으로 구성·운영되어짐으로 시험·측정단계에서 현실적인 부하가 반영되어 평가되어질 수 있도록 한다.

4장에서 제시되어지는 사례는 다종의 Email 서버가 어느 레벨까지의 부하를 효율적이고 생산적으로 처리하는 지를 파악하고자 하는 의도로 설계되었다.

3.3. 계획 수립 및 문서화 단계

소프트웨어 벤치마크 테스트 계획 수립을 문서화 하는데 있어 경험적 주요 항목은 다음과 같다.

[표 4] 테스트 계획 수립 주요 항목 및 지침

항목	비고
테스트 목적	· 벤치마크 테스트를 통해 달성하려는 사항 (제품 간 우선순위 매김, 장·단점 파악, 이전 버전과의 비교 등)과 배경(선정근거)를 기술
테스트 범위	· 테스트 대상 시스템 컴포넌트 및 S/W
평가 항목	· 주어진 테스트 범위 내 모든 성능 요소에 대한 비교·평가가 이상적이나 시간·비용에 대한 현실적 한계로 반드시 수행해야 하는 항목과 그렇지 않은 항목으로 우선순위 매김 (H,M,L 등) 가능 · 적절한 판단 기준은 평가기준으로 대신함
테스트 완료조건	· 테스트의 성공/실패/재-수행조건을 명확하게 기술하여 테스트 자원 소모 등에 대한 문제점을 사전에 방지 · 테스트 조직에 속한 모든 사람들은 사전에 성공/실패기준을 반드시 숙지하고 있어야 함
테스트 스크립트 명세	· 성능 평가 (세부)항목에 대한 테스트에 사용되어지는 스크립트 형태를 기술 · 필요 스크립트 형태 및 빈도수는 사용자 프로파일에 의해 결정됨 · 대부분의 사용자 프로파일은 몇 개의 테스트 스크립트의 패턴(상대적 빈도수 또는 입력데이터 파라미터)에 의해 결정되는 경향이 있음 (예) 로그인(1)→메일검색(1)→메일발송(3)→로그아웃(1)
테스트 시나리오 명세	· 성능 벤치마크 테스트 시나리오를 기술하기 위해 스크립트가 결합되는 단계 · 기본적으로 스크립트, 부하레벨, (Ramp Up/Ramp Down 시간)을 포함함. · 서로 다른 부하레벨을 시뮬레이션 하기 위해 다중 시나리오를 계획할 수 있음 (각 시나리오는 각각의 명확한 테스트 목적 및 완료조건을 가지고 있어야 함.) · 다양한 기준에 의해 복합적으로 측정·평가되어도 부하레벨에 대한 신뢰성 있는 지표가 사용되어야 함. 신뢰성 있는 지표란 소프트웨어 벤치마크 테스트 결과와 관련된 고객, 시험원, 평가자들 간 이견 없이 동의

	할 수 있는 결과로 객관적으로 신뢰할 만 하여야 함
평가기준	<ul style="list-style-type: none"> · 평가기준 선정은 크게 제품에 대한 비즈니스 요구사항과 평가자 요구사항에 기반 함 · 소프트웨어가 요구사항을 만족하는지를 확인하기 위하여 품질 평가기준(Criteria)을 정량화하기 위한 메트릭을 기술 · 테스트 결과에 대한 객관성과 신뢰성 확보를 위해 각 Attribute에 대해 주어지는 수치와 범위에 대해 S/W벤치마크 테스트 관련자간(고객, 테스트 조직, 평가자 등)의 합의 절차가 필요 · Measurement는 유효하고 적절한 오차한계 내에서 재생성 가능해야 하며 다수 제품 간 또는 기준 값과의 신뢰할만한 비교 (comparisons)를 위해 엄격한 Metrics의 사용이 필요(다수의 소프트웨어 벤치마크 테스트 수행을 위한 즉, Measurement scale을 범주화하기 위해서는 최소 순차적 (Ordinal) Scale 이상의 메트릭이 요구됨)
측정항목 및 방법	<ul style="list-style-type: none"> · 평가 메트릭을 구성하는 각 요소 데이터와 수집방법을 기술 · 각 항목은 충분한 정확성을 가지고 측정 가능하여야 함
테스트 고려사항	<ul style="list-style-type: none"> · 테스트 수행 시 특별히 주의가 요구되는 H/W·S/W 설정, 필요 리소스(인적 자원, 자동화 툴, 각종 테스트 비교평가 데이터) 등을 기술 · 중요 요소를 기술함으로써 테스트 수행 전 필요 요건을 재점검

4. 사례연구

4.1 사례 개요 및 방법

이 장에서는 본 논문에서 제시된 주요 항목 및 지침을 '상용 OS의 SMTP/POP 서버성능 벤치마크 테스트' 계획 수립에 대한 사례연구를 수행함으로써 그 신뢰성 및 유용성을 검증하고자 한다.

4.2 사례 및 결과

4.2.1. Email 서버 벤치마크 테스트 계획 수립

다음 [표 5]~[표 6]은 본론에서 제시된 주요 항목 및 지침을 반영하여 작성한 테스트 계획 사례를 나타낸다.

[표 5] 테스트 계획 수립 주요항목 사례

항목	비고												
테스트 목적 및 선정근거	· SMTP/POP서버는 Email 서버 중 가장 범용적으로 사용되는 서버로써 서버 OS의 이용목적 중 비중이 높은 항목으로 테스트 대상 제품간의 장·단점 파악												
테스트 범위	· 동시 사용자에게 대한 Email 서비스의 성능												
평가항목 및 내용	<ul style="list-style-type: none"> · 평가항목 <ul style="list-style-type: none"> - 동시 접속자 변화에 따른 서버 응답속도 - 동시 접속자 변화에 따른 서버 처리능력 · 평가내용 <ul style="list-style-type: none"> - 시스템 효율성(평균 CPU 사용율 등) - 동시 접속자별 서버 평균 응답 속도 및 메시지 처리능력 (Server Productivity) 												
평가기준 (Metric) 및 측정항목	<ul style="list-style-type: none"> · 평가기준 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="background-color: #cccccc;">메트릭 이름</th> <th style="background-color: #cccccc;">메트릭(측정항목)</th> <th style="background-color: #cccccc;">단위</th> </tr> </thead> <tbody> <tr> <td>서버 응답시간</td> <td>$X = (\text{전체 서버 응답시간의 합}) / (\text{총 Transaction Request의 수})$</td> <td>sec</td> </tr> <tr> <td>메시지 처리능력</td> <td> $X = Y/Z$ $Y = \text{시험 기간동안 총 송신한 메시지의 수}$ $Z = \text{시험 기간동안 총 수신한 메시지의 수}$ </td> <td>%</td> </tr> <tr> <td>CPU 사용율</td> <td> $X = \sum(A_i)/T$ $A = i\text{번째 측정 시 CPU사용율}$ $T = \text{측정 시간}$ </td> <td>%</td> </tr> </tbody> </table>	메트릭 이름	메트릭(측정항목)	단위	서버 응답시간	$X = (\text{전체 서버 응답시간의 합}) / (\text{총 Transaction Request의 수})$	sec	메시지 처리능력	$X = Y/Z$ $Y = \text{시험 기간동안 총 송신한 메시지의 수}$ $Z = \text{시험 기간동안 총 수신한 메시지의 수}$	%	CPU 사용율	$X = \sum(A_i)/T$ $A = i\text{번째 측정 시 CPU사용율}$ $T = \text{측정 시간}$	%
메트릭 이름	메트릭(측정항목)	단위											
서버 응답시간	$X = (\text{전체 서버 응답시간의 합}) / (\text{총 Transaction Request의 수})$	sec											
메시지 처리능력	$X = Y/Z$ $Y = \text{시험 기간동안 총 송신한 메시지의 수}$ $Z = \text{시험 기간동안 총 수신한 메시지의 수}$	%											
CPU 사용율	$X = \sum(A_i)/T$ $A = i\text{번째 측정 시 CPU사용율}$ $T = \text{측정 시간}$	%											
평가방법 및 측정방법	<ul style="list-style-type: none"> · 평가방법 <ul style="list-style-type: none"> - 1명의 사용자가 미리 정의된 사용자 프로파일을 처리하는데 소요되는 서버 응답시간을 나타냄, 각 트랜잭션 가중치 평균을 이용하여 계산되며 수치가 낮을수록 높은 성능을 나타냄 - 메시지 처리능력 : 정해진 시간동안 얼마나 많은 메시지를 전송하고 수신하였는지를 측정할 수치로서 서버의 작업능력을 나타냄. 서버의 작업능력은 총 송신된 메시지의 수 대비 수신된 메시지의 수의 비율로 계산되며 높은 수치가 우수한 서버 작업능력을 보임 · 측정방법 <ul style="list-style-type: none"> - XXX 성능 시험도구를 사용하여 신뢰할만한 성능 데이터를 획득할 정도의 기간동안 테스트를 지속하여 수행 - 시스템 자원 측정도구 XXX를 이용하여 서버 OS의 성능관련 데이터를 획득 및 분석 수행 												
테스트 고려사항	<ul style="list-style-type: none"> · 각 테스트 수행 후, 시스템 재부팅 등의 시스템 안정화 기간 확보 · 결과의 공정성 및 신뢰성 확보를 위해 테스트를 반복 수행 결과들의 평균값 이용 												

<ul style="list-style-type: none"> · 각 벤치마크 테스트 대상 제품들 간의 테스트베드를 동일하게 구성(본 논문에서는 테스트 베드 생략) · Email 서버의 주요 프로토콜(SMTP/POP)에 대한 테스트 시나리오를 다양화

• 테스트 시나리오 명세

본 사례연구에서 사용된 테스트 시나리오 및 부하레벨은 아래와 같이 구성되었다.

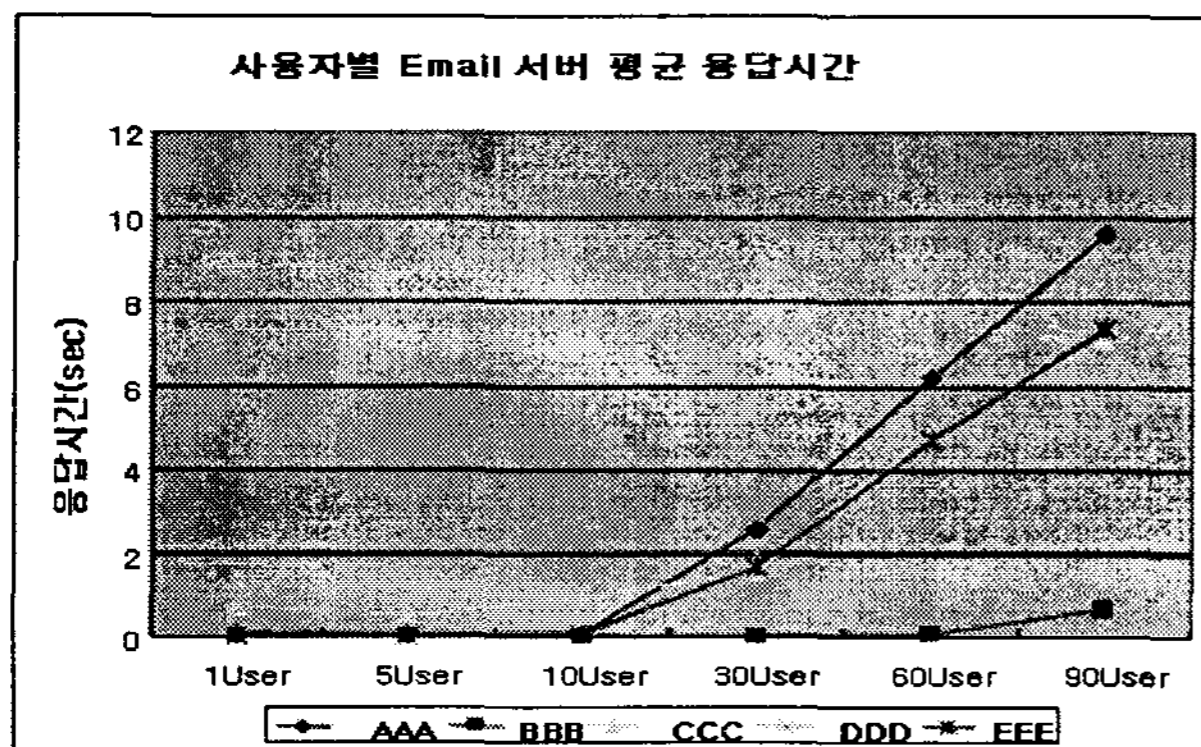
- 프로토콜(SMTP/POP)별 사용자 프로파일
- 프로토콜(SMTP/POP)별 사용자 요청비율
- 동시 서비스 요청자의 수
- 동시 서비스 요청자수별 테스트 시간
- 전송 메시지 용량
- 전송 메시지의 동시 수신자 수

[표 5] Email 서버 벤치마크 테스트 시나리오 및 부하레벨

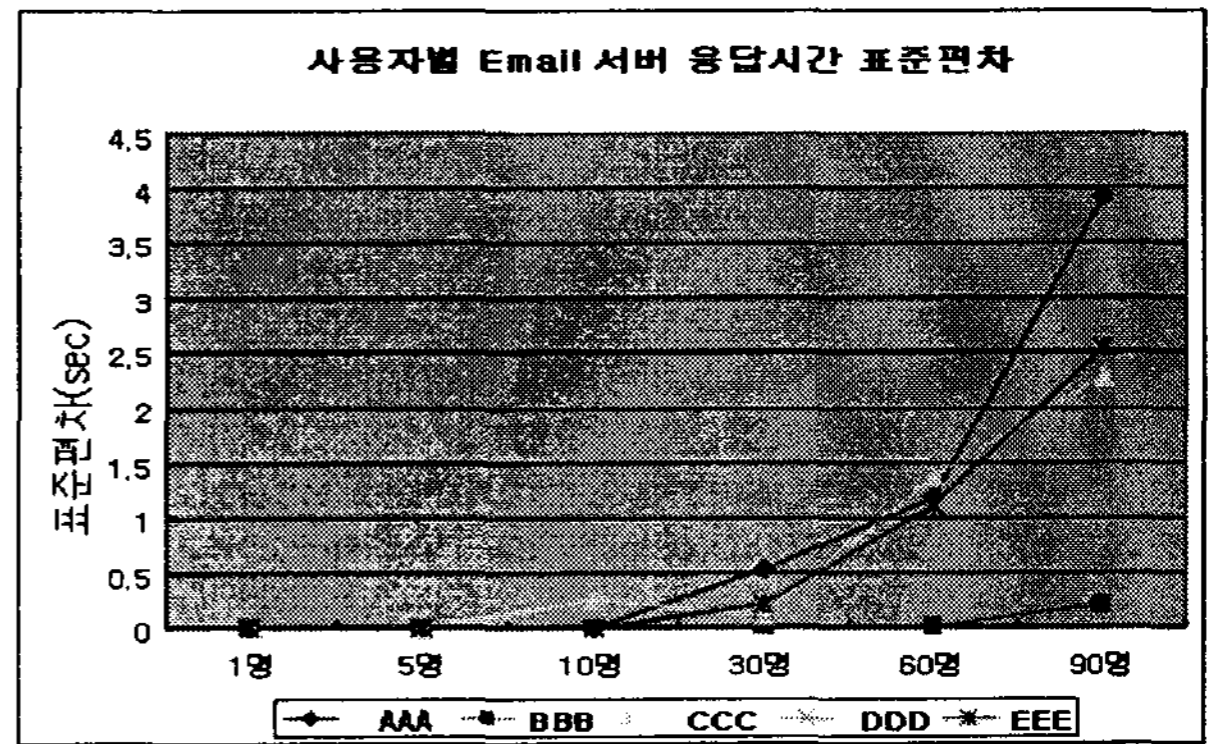
동시 사용자 수	사용자 프로파일 및 요청비율 (프로도형별)		전송 메시지 용량				각 유저별 시험 기간	비고
	작업비율	이동첨부파일	용량 구분	1K	4K	8K		
1명	-로그인	작업비율에 따라	수신자 수	1	1	1	1	• 서버응답시간에 따라 총 수행횟수 및 실행 시험시간은 다름
5명	-수신 메시지 수 확인	이동첨부파일	작업 비율	12	6	8	3	
10명	-메일박스 지움	정해진 수신자에						
30명	-메일박스 지움	게 전송함						
60명	-로그아웃							

4.2.2. Email 서버 벤치마크 테스트 주요 결과

다음 [그림 4]~[그림 6]과 [표 6]은 4.2.1에서 수립된 계획을 기반으로 5종의 Email 서버에 대한 소프트웨어 벤치마크 테스트 수행결과를 보인다.

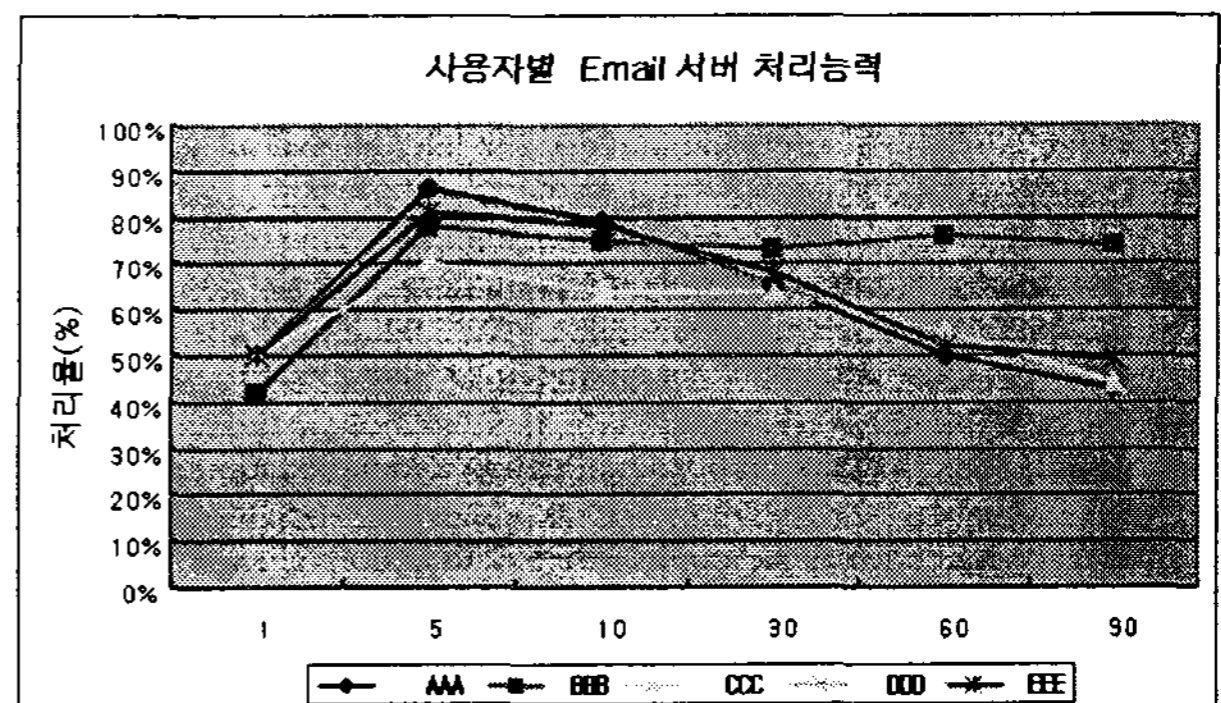


[그림 4] 제품별/사용자별 Email 서버 평균응답시간



[그림 5] 제품별/사용자별 Email 서버 평균응답시간

- BBB는 동시 접속자 수에 따라 평균응답시간이 거의 차이가 없으며, 타 제품대비 월등한 "Email 서버의 평균 응답시간"을 보였음
- AAA의 경우, 동시 접속자 수 1명 ~ 30명까지 타 4개 제품과 비슷한 수치를 보이다 30명 이후 "평균 응답시간"의 급격한 증가를 보임
- CCC, DDD, EEE의 경우 미세한 차이는 있으나 비슷한 성능추이를 보였음

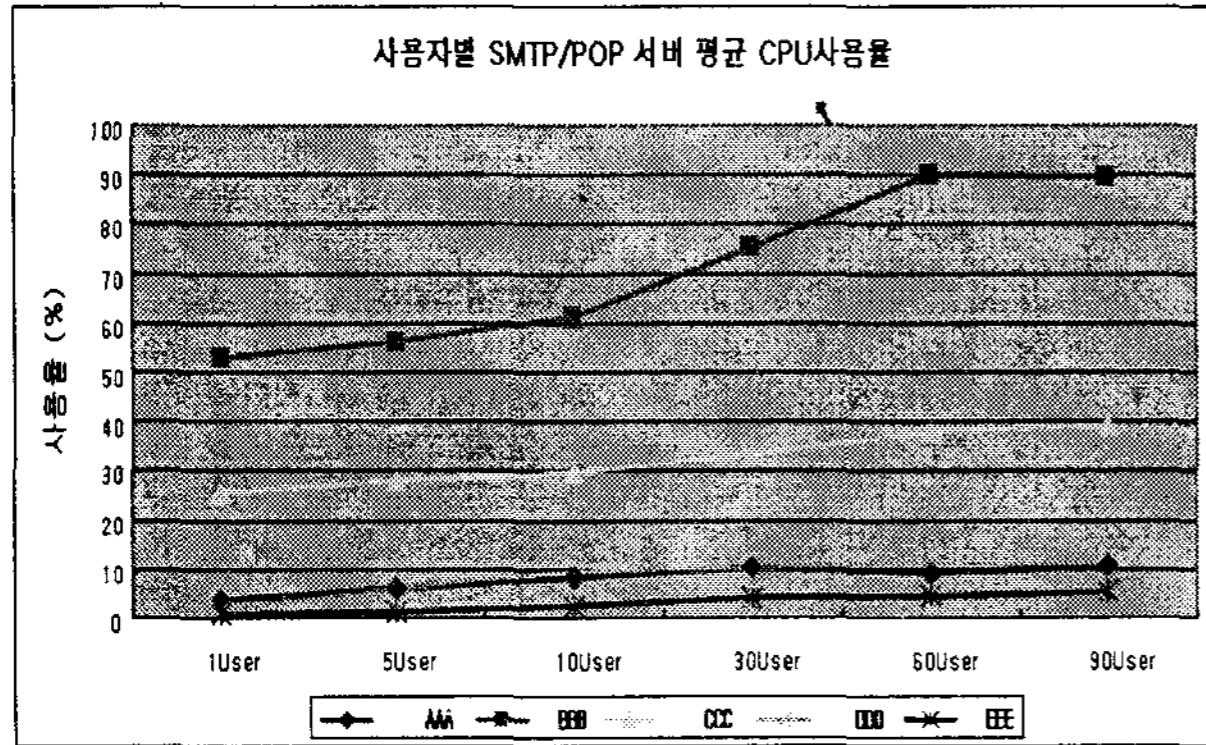


[그림 6] 제품별/사용자별 Email 서버 처리능력

[표 6] 제품별/사용자별 Email 서버 처리능력 측정결과

제품	AAA		BBB		CCC		DDD		EEE	
	송신 메시지	수신 메시지	송신 메시지	수신 메시지	송신 메시지	수신 메시지	송신 메시지	수신 메시지	송신 메시지	수신 메시지
1명	6	3	7	3	4	2	6	3	6	3
5명	36	31	37	29	30	21	35	30	37	30
10명	72	57	74	56	51	33	73	57	73	57
30명	98	64	225	165	132	85	115	75	124	85
60명	104	52	430	328	134	75	119	62	129	68
90명	97	42	479	321	123	55	111	55	117	58

- 동시 접속자 수 1명 ~ 30명 구간에서는 모든 제품이 비슷한 서버 작업능력을 보였다
- 동시 접속자 수 30명 이후부터는 BBB가 월등한 서버 작업능력을 보였다
- 서버 응답시간 및 서버 작업능력 모두 고려시, 타 제품 대비 BBB의 SMTP/POP 서버 능력이 상대적으로 우수함



[그림 7] 제품별/사용자별 Email 서버 평균 CPU 사용률

- BBB는 타 제품대비 매우 높은 CPU사용율을 보였으며 타 제품의 경우 평균 CPU 사용율의 변화가 크지 않음

5. 결론 및 향후 연구

본 논문에서는 기존의 소프트웨어 품질 평가 프로세스 및 성능 테스트 기술을 검토·분석하고 경험적 지식에 기반을 두어 성능측면의 소프트웨어 벤치마크 테스트 계획을 수립하는데 있어서의 주요 항목과 관련 지침을 제시하였다.

이는 정확한 수치가 나오는 하드웨어 및 네트워크 제품과는 달리 소프트웨어에 대한 평가가 어렵고 객관적인 입장에서 수행되어야 하며 신뢰할 만한 결과 도출방안이 마련되어야 한다는 벤치마크 테스트 이슈사항에 착안하여 테스트 초기단계에서부터 조기에 관련 문제점과 위험요소를 식별하여 해결할 수 있는 효율적인 테스트 계획 수립 기반을 마련하고자 함이다.

또한 'Email 서버의 성능 벤치마크 테스트'를 통한 사례연구를 수행함으로써 유용하게 적용되어 질 수 있음을 보였다.

그러나 전체적인 소프트웨어 성능 벤치마크 테스트 프로세스를 다루지 못하고 다수의 사례에 맞게 유연하게 적용할 수 있는 구체적 방안이 부족하다.

따라서 향후 연구과제로는 테스트 전략, 테스트 데이터의 선택, 성능 목표 및 성능 테스트 방법에 대한 연구와 많은 사례연구를 통하여 정제해 나가야 할 것이며 소프트웨어 벤치마크 테스트에 필요한 성능 도구의 개발과 방법론에 대한 연구를 지속적으로 수행해야 한다.

[참고문헌]

- [1] Connie U. Smith, Performance Engineering of Software Systems, Addison-Wesley Publishing Company, 1990
- [2] Rational Corporation, Rational Performance Engineering White Paper, <http://www.rational.com>
- [3] Setting Performance Standards: Concepts, Methods, and Perspectives by Gregory J. Cizek, Gregory L. Cizek
- [4] Methods and Quantitative Techniques for Proactively Managing System Performance, Seminar by Dr. Connie U. Smith, November 4-7, 2002, Santa Fe, NM
- [5] Automated Software Testing : Introduction, Management, and Performance, Elfriede Dustin, Jeff Rashka, John Paul, Addison-Wesley
- [6] Software Engineering Body of Knowledge, IEEE Computer Society, 2001
- [7] The Unified Software Development Process, IVAR JACOBSON, GRADY BOOCH, JAMES RUMBAUGH, Addison Wesley 1999
- [8] 네트워크기반 침입탐지시스템의 성능 테스트 모델, 정도균, 숭실대학교, 2002
- [9] S/W분야별 벤치마킹 테스트 모델 개발 및 시범서비스에 관한 연구 보고서, 한국정보통신기술협회(TTA), 2002
- [10] 벤치마킹 테스트 보고서, 한국정보통신기술협회(TTA), 2004