

## OLTP서버 규모산정방식의 개선에 관한 연구\*

나종희\*\* · 최광돈\*\*\*

### A Study on Improvement of OLTP Server Sizing Method\*

Jonghei Ra\*\* · Kwangdon Choi\*\*\*

#### ■ Abstract ■

Hardware sizing is determined by an increasing rate of operated business and characteristic of adapted technology which are imported systems. Because it is very difficult for intuitive decision of hardware sizing's appropriateness. But in spite of difficulty, for maximizing the utilization of information resource and the effect of information investment in public sector are need to correctness hardware sizing. In this paper, we proposed hardware sizing method for OLTP server that is make up it's faults of proper proposal method. Our study certainly will act as a catalyst for higher investment-efficiency of future information systems in public sector.

Keyword : OLTP(On-Line Transaction Processing), Server Sizing, TPC-C, tpmC

논문투고일 : 2009년 07월 19일      논문수정완료일 : 2009년 09월 06일      논문게재확정일 : 2009년 09월 12일

\* 이 연구는 2009년도 광주대학교 대학 연구비의 지원을 받아 수행되었음.

\*\* 광주대학교 e-비즈니스학과 부교수

\*\*\* 한세대학교 경영학부 부교수

## 1. 서 론

정보화 시대의 도래에 따라 사회적으로 정보인프라 구축에 대한 관심이 고조되고 있으며, 효율적인 업무처리, 고객 서비스 개선 등을 위해 다양한 정보시스템이 구축되고 있다. 이러한 정보시스템은 일반적으로 하드웨어, 소프트웨어, 인력 등으로 구성되며, 이 중 하드웨어는 공급업자 혹은 시스템 구축자 등에 따라 규모산정 적용항목 및 적용비율을 경험적으로 적용하기 때문에 부정확한 규모가 산정되는 경우가 많이 발생한다[1, 2].

하드웨어 규모는 업무의 성격, 업무 증가율, 사용자 사용빈도, 구축기술 등을 전체적으로 고려하여 산정해야 하므로, 시스템 구축사업에서 하드웨어 규모 적정성의 옳고 그름을 판단하는 것은 어려운 일이다. 시스템 구축사업에서 하드웨어가 차지하는 비중이 전체 프로젝트 비용의 적게는 50%에서 많게는 90%를 차지함에도 불구하고, 그 동안 하드웨어 규모산정 부분은 사업자나 장비업체에 전적으로 의존해 왔으며, 소프트웨어 개발에 비하여 상대적으로 소홀한 분야로 많은 관심을 갖지 않았다. 이로 인해서 실제 요구되는 하드웨어의 각 구성요소가 사업자나 장비업체에 의해 과다 또는 과소 산정되는 경우가 발생하여도 마땅히 개선할 수 있는 방법이 없었다[4, 5].

하드웨어 규모산정은 도입되는 시스템을 운영할 업무의 성격과 업무의 예상증가율, 구축기술의 특성들을 전체적으로 고려하여 산정해야 하므로 규모산정의 적정성을 직관적으로 판단하는 일은 매우 어려운 일이다. 따라서 이러한 어려움에도 불구하고 공공부문 정보화자원의 효율적 운영을 통한 정보화 투자효과의 극대화를 위해서 한국정보화진흥원(舊, 한국정보사회진흥원)에서는 그 동안의 연구결과를 정리하여 2008년 12월 정보통신단체표준(TTA Standard)으로 “정보시스템 하드웨어 규모산정 지침”을 제정하였으며, 이에 근거하여 공공부문 정보화사업에서의 하드웨어 규모산정이 이루어지고 있는 상황이다[8].

본 연구는 이러한 정보시스템 하드웨어 규모산정 방식 중 OLTP서버 산정모델에 대한 개선의 일환으로써 보다 정확한 규모산정을 위하여 기존 지침이 가지고 있는 문제점을 파악하고 개선된 모델을 제시하는데 그 목적이 있다. 이를 위해 정보시스템 하드웨어 규모산정 지침과 기존 규모산정 연구를 기초로 하여 문제점에 대한 분석과 개선 모델을 수립하고 관련 전문가의 검토(탐색적 사례 연구 및 전문가 검토) 과정을 통해서 최종적인 산정방식을 제시하였다.

## 2. 이론적 배경

### 2.1 규모산정 개념

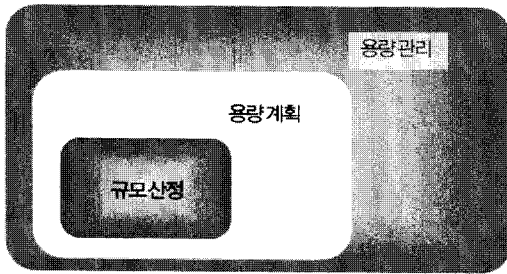
일반적으로 용량과 관련되는 용어로 용량관리, 용량계획, 규모 혹은 용량산정을 혼용하여 사용하고 있다. 그러나 이들 용어간의 차이를 명확히 구분하기는 쉽지 않지만 이들 세 가지 개념을 정리하면 다음과 같이 정의 할 수 있다[6, 7].

첫째, 용량관리는 업무요구사항을 충족시키기 위한 현재와 미래의 자원에 대한 용량계획을 수립하고 비용과 용량의 균형을 맞추기 위한 관리행위이다. 따라서 용량관리의 대상은 시스템, 네트워크 등 조직 내의 하드웨어 자원만을 국한하는 것이 아니라 전사적인 자원을 관리 대상으로 하며, 지속적인 관리에 중점을 둔다.

둘째, 용량계획은 개략적인 시스템 구조와 응용 업무를 기반으로 시스템이 요구하는 성능을 결정하기 위한 과정이다. 용량계획에서는 클라이언트 어플리케이션의 형태, 동작 특성, 이들에 접근하는 사용자의 수, 서버시스템에 대응하는 오퍼레이션의 형태, 서버시스템에 접속하는 동시 접속자 수, 서버시스템에 의해서 수행되어야 하는 피크율, 피크타임 하에서의 여유율 등을 결정한다.

셋째, 규모산정은 “기본적인 용량과 성능요구사항이 제시되었을 때, 그것을 시스템 요구사항으로 변환하는 것”을 말한다. 즉, 시스템 규모산정이 실

제 업무와 응용을 기반으로 수학적인 방법론을 사용하여 도입하고자 하는 시스템의 규모를 추정 혹은 계산하는 것이다.



[그림 1] 용량 관련 개념간의 관계

따라서 개념적인 차이에 따른 용량관리, 용량계획 그리고 규모산정간의 관계는 [그림 1]에서와 같으며, 본 연구에서는 조직적인 차원에서 지속적으로 이루어지는 용량계획이나 용량관리 보다는 시스템 단위에서 일시적으로 이루어지는 규모산정에 대해 다룬다.

한편, 규모산정 방식은 수식계산법(Calculating Method), 참조법(Referencing Method), 시뮬레이션법(Simulation technique) 등 3가지로 구분할 수 있다[1].

첫째, 수식계산법은 사용자 수 등 규모산정을 위한 요소를 토대로 규모수치를 계산하고, 보정치를 적용하는 방법으로서 규모산정의 근거를 명확하게 제시할 수 있으며, 참조법, 시뮬레이션법 등 다른 방법에 비해 비교적 간단하게 규모를 예측할 수 있다는 장점을 갖는다. 이 방법의 단점으로는 보정치가 잘못되었을 경우 원하는 값과 많은 차이가 발생하며, 보정치에 대한 정확한 근거자료 제시가 어렵다는 특성을 갖는다.

둘째, 참조법이다. 이러한 참조법은 구축하고자 하는 시스템의 사용자수, DB 크기 등 업무량에 대한 기본 데이터를 토대로 비슷한 규모의 이전시스템과 비교 최종적인 규모를 추정하는 방식이다. 이 방식은 기존 구축되어 있는 업무 시스템과 비교가 가능하므로 비교적 안전한 규모산정이 가능

하다는 장점을 갖는데 비해, 단점으로는 유사모델이 존재하지 않는 경우 정확한 추정이 어렵고 계산에 의한 방법이 아닌 비교에 의한 것이므로 산정의 근거를 명확히 제시하기 어렵다.

셋째, 시뮬레이션방법이다. 이는 대상업무에 대한 작업부하를 모델링하고 시뮬레이션하여 규모를 산정하는 것으로 일반적으로 사전 벤치마크를 하는 방식이다. 앞서 설명한 계산법이나 참조법에 비해 상대적으로 정확한 값을 얻을 수 있다는 장점이 있으나 시뮬레이션 환경을 만들고 시험하는데 많은 시간과 노력이 소요된다는 단점을 갖는다.

이러한 규모산정은 예측의 문제로 산정의 정확성은 산정방식에 큰 영향을 받는다. 따라서 정확한 규모산정을 위해서는 주어진 여건을 고려하여 적절한 산정방식을 적용하는 것이 필요하다.

## 2.2 서버 성능평가

컴퓨터가 발명된 이후, 일반적인 컴퓨터의 성능 측정을 위한 단위로 초당 명령어 처리개수를 나타내는 MIPS(Million Instruction Per Second)가 주로 사용되어왔다. 그러나 이러한 MIPS는 OS가 존재하지 않았던 하드웨어상에서 업무용 프로그램을 직접 실행하던 상황에서 적합한 것으로 오늘날과 같은 비즈니스 환경에서 더 이상 이를 활용하기 어렵게 되었다[1].

따라서 이러한 문제를 해결하고자 1980년대 중반 하드웨어 및 DBMS 제조사를 중심으로 하여 TPC, SPEC 등 국제적인 비영리 성능평가 기관이 설립되었으며, 이들 기관에서는 업무특성에 따른 성능평가 기준을 마련하고 이에 근거한 성능측정을 실시하게 되었다.

이들 성능측정기관에서 제시하고 있는 기준은 <표 1>에서와 같이 업무유형이나 작업부하형태, 그리고 시스템아키텍처에 따라 벤치마크 기준을 제시하고 있다. 이러한 벤치마크 기준은 전문가 그룹에 의해 오랜 작업을 거쳐 만들어지며, 하드웨어기술의 진보와 비즈니스 작업부하 형태 변화

등의 이유로 기존 기준들이 폐기되거나 새로운 벤치마크 기준으로 대체된다.

〈표 1〉 기관별 대표적인 성능평가기준

기관	성능평가 기준	적용 업무 유형
TPC [10]	TPC-C	OLTP 중심 어플리케이션, OLTP 및 배치 어플리케이션
	TPC-E	OLTP 중심 어플리케이션
	TPC-H	의사결정지원 어플리케이션
	TPC-APP	웹 혹은 웹어플리케이션
SPEC [11]	SPECjbb2005	웹 어플리케이션
	SPECWeb 2005	웹
	SPECjAPPServer 2004	웹 어플리케이션

현존하는 대표적인 벤치마크로는 대체로 OLTP 성 업무 처리 서버의 성능측정에 강점을 가지고 있는 TPC의 경우, TPC-C와 TPC-H와 최근에 발표된 TPC-E 등이 있다. TPC-E는 1980년대 중반에 발표된 TPC-C 문제를 해결하고자 TPC-C의 후속모델로 2006년에 발표되었으며, 여러 가지 시장 상황으로 TPC-C가 폐기되지 않고 현재 TPC-E와 공존하고 있다. 한편, SPEC의 경우 웹(Web)이나 웹 어플리케이션(Web Application) 서버의 성능측정을 위한 벤치마크가 주류를 이루며, 대표적인 벤치마크로는 SPECWeb2005, SPECjAPPServer2004, SPECjbb2005 등이 있다.

TPC, SPEC는 벤치마크 기준에 따라 측정된 서버의 성능치를 일정기간(분기 등)별로 발표하고 있다. 이러한 성능치에 대한 평가는 성능측정기관이 직접 평가하는 것으로 생각하기 쉬우나 제조사와 감리전문가(Auditor)를 통한 간접평가 방식을 사용한다. 따라서 TPC나 SPEC은 일반적으로 알려진 성능측정기관이라기 본다는 성능인증기관으로 부르는 것이 타당할 것이다. 한편, 성능치의 평가 및 발표는 다음과 같은 절차에 의해서 이루어진다.

우선, 하드웨어 제조사가 성능치를 인증 받고자 하는 서버장비에 대해 규정된 벤치마크를 수행하고 그 결과를 성능측정기관에 보낸다. 성능측정기

관은 그 결과를 사전에 인증한 해당 벤치마크 감리전문가에게 보내 감리(Audit)를 수행한다. 감리전문가는 하드웨어 제조사가 성능인증을 받고자 하는 제품에 대한 성능평가가 벤치마크 기준에 의거하여 적절한 절차에 의해서 이루어졌는가를 평가하여 그 결과를 성능인증기관에 보낸다. 성능인증기관은 그 결과를 토대로 기관내의 상설기구에 의한 검토를 거쳐 최종적으로 확정 발표한다.

이러한, 하드웨어 제조사에게 있어서 인증기관에 의한 서버장비의 성능인증은 오랜 시간과 비용이 소요는 작업이다. 이로 인해서 하드웨어 제조사가 생산하는 모든 제품군에 대해서 성능인증을 받기는 현실적으로 매우 어려운 일이다. 더욱이 동일모델에 해당하는 장비의 경우에도, 채용 CPU개수, 외장 디스크 등 구성요소나 방식에 따라 다른 성능을 보이게 된다. 또한, 자사의 제품의 성능을 자랑하기 위해서 현실에서 활용이 불가능한 사양으로 시험 환경을 마련하고 이에 대한 성능을 제시하기도 한다.

이로 인해, 성능인증기관에서 제시하고 있는 벤치마크기준의 벤치마크 환경이 실제 운영환경을 모사할 수 있는지에 대한 일반적인 논란을 그냥 치더라고 성능치의 신뢰성과 객관성, 현실성에 대한 논란은 여전히 대두되고 있다.

### 2.3 규모산정 및 성능평가에 관한 선행연구

서버 규모산정 측면에서 관련 연구는 소프트웨어 개발 분야에 비해 연구가 매우 미약한 상황이다. 규모산정의 대표적 연구로는 나중희[2], 정해용[4]의 연구가 있다. 나중희[2]의 연구는 국내 공공부문 정보화사업의 사례와 국내 SI업체의 산정기준을 토대로 하여 정보시스템 구축비용 중에서 가장 중요한 CPU, 메모리, 디스크의 규모산정 방식과 절차를 제시하고 있으나 선행연구가 거의 없는 탐색적 차원의 연구들로 연구의 객관성 부족으로 인해 신뢰도와 타당성 측면에서 일반적인 이론으로 해석하는 데는 한계점으로 지적되고 있다.

정해용[4]의 연구는 나종희[2]의 연구를 선행 연구로 하여 이론적 문헌검토와 SI업체 및 하드웨어 공급업체 규모산정 전문가들의 경험적 노하우 등을 통하여 기존에 활용되고 있는 규모산정 방식에 근거하여 기준이론을 설정하고 이를 현장의 전문가들을 대상으로 설문조사 및 전문가 집단토의 방식을 통하여 타당성을 검증하는 작업을 수행하였다.

또한, 국가적인 차원의 연구로는 한국정보화진흥원의 연구를 들 수 있는데, 한국정보화진흥원에서는 국가정보화사업에 적용할 수 있는 규모산정의 기준을 제시하고자 2003년도부터 2006년까지 약 4년 간에 걸쳐 위의 연구자들과 하드웨어 규모산정에 관한 연구를 수행하였으며, 이러한 연구 결과를 토대로 2008년 12월, TTA의 정보통신단체표준으로 “정보시스템 하드웨어 규모산정 지침”을 제정하였다.

한편, 성능평가측면에서는 앞서 제시한 다양한 성능 벤치마크 기준의 직접적인 활용보다는 공공부문을 중심으로 도입 서버의 성능을 규정하기 방법으로 도입장비의 제안요청이나 제안 시에 TPC-C의 tpmC 값과 같은 벤치마크의 성능 측정단위를 활용하고 있으며, 도입이후 일부 벤치마크를 따른 직접적인 성능측정을 위한 기준으로 활용하려는 시도는 있으나 이보다는 제안요청서나 제안서상에 명시된 성능의 충족여부를 확인하는 차원에서 성능인증기관에서 발표한 서버별 성능측정치 활용하고 있다.

#### 2.4 OLTP서버 성능측정을 위한 벤치마크 : TPC-C

TPC-C는 1980년대 만들어진 OLTP서버 성능측정을 위한 대표적인 벤치마크이다. 그러나 이러한 TPC-C는 현대적인 업무환경에 부합하지 않는다는 다양한 지적에 따라 TPC는 TPC-C의 후속 모델로 2006년 후속모델로 TPC-E를 발표하였다. 일반적으로 새로운 후속모델이 발표되면, 기존 모

델이 퇴역(retired)하는 것과 달리 시장요구에 의해서 TPC-C는 TPC-E와 현재까지 양립하고 있는 상황이다. 본 절에서는 이러한 OLTP 서버 성능 측정의 대표적인 TPC-C, TPC-E중 규모산정의 기준으로 사용되고 있는 TPC-C를 살펴보고자 한다.

TPC-C의 서버성능 측정치는 tmpC로 표시하는데, 이것은 분당 처리하는 New-Order의 “비즈니스 성취도”를 측정한다. 즉, DB 서버 등 OLTP 성업무를 수행하는 서버의 tpmC를 측정하기 위해서는 <표 2>에서와 같이 DB수준의 트랜잭션이 아닌 다섯 가지 종류의 비즈니스 트랜잭션을 각 비율대로 동시에 발생시킨 후 안정적인 상태(Sate)에서 New-Order 트랜잭션 분당 처리건수를 의미한다.

<표 2> 트랜잭션별 최소발생 및 허용응답시간

구 분	최소 발생율	허용응답시간
New-Order	N/A	5초 이내
Payment	43%	5초 이내
Order-Status	4%	5초 이내
Delivery	4%	5초 이내
Stock-Level	4%	20초 이내

이러한 성능측정 결과는 각 트랜잭션이 허용응답시간 내에 90% 이상 수행 되었을 때 인정한다. 여기에서의 응답시간은 서버차원의 응답시간이 아닌 클라이언트 차원의 응답시간을 지칭한다.

한편, TPC-C는 어떠한 특정 비즈니스 영역의 활동을 나타낸 것이 아니고 제품이나 서비스의 관리나 판매, 분배 등을 구현한 것으로 실제 응용을 어떻게 만들 것인가를 제시하고 있지는 않다. 벤치마킹의 목적은 응용시스템의 성능특성(적정수준의 시스템 활용도, 운영의 복잡도)을 유지하면서 사용자들에게 정확하고 객관적인 성능자료를 제공하기 위한 것으로 TPC에서 구상한 TPC-C 타입의 트랜잭션들은 일반 소매상을 기준으로 실제 소매 업무를 수행하기 위해 필요한 모든 트랜잭션들

을 모델화한 것이다.

### 3. 규모산정 개선 모델

#### 3.1 기존 모델 분석 및 신규 모델 수립 방법

본 연구는 문헌조사와 사례연구를 병행한 탐색적 연구와 전문가집단의 검증을 통한 연구를 수행하였다. 이러한 사례연구 방법은 초기단계의 학문적 연구에서 이론을 생성하거나 기존 연구에 새로운 시작을 위하여 유용한 방법이다. 국내외의 정보시스템에 대한 규모산정에 대한 연구는 한국정보화진흥원을 중심으로 제한적으로 이루어지고 있어 이론적인 토대가 매우 약한 상황이다. 따라서 본 연구에서는 한국정보화진흥원이 TTA 정보통신단체표준으로 제정한 “정보시스템 하드웨어 규모산정 지침”과 기존 연구자의 연구결과를 심층 분석하고 규모산정에 대한 현행 산정식이 갖는 문제점과 개선방안을 제시하였다. 다음으로 분석결과에 대해서 규모산정 전문가 집단토의방식인 표적 집단면접법(Focus Group Interview, FGI)을 수행하여 규모산정방식과 규모산정식에 포함된 항목

및 항목별 입력값의 범위를 조정하는 등의 연구방법론을 채택하였다.

#### 3.2 기존 규모산정 모델 분석

OLTP서버의 CPU에 대한 규모는 <표 3>에서 보인바와 같이 분당트랜잭션수, 기본 tpmC 보정, 시스템여유율 등 8개 산정항목에 대한 산술적 곱에 의해서 계산된다. 이들 규모 산정항목 중규모산정에 가장 큰 영향을 미치는 항목은 tpmc의 정의에서 알 수 있듯이 “분당트랜잭션 수”이다. 따라서 분당 트랜잭션의 정확한 추정치는 결국 규모산정결과의 정확성과 직결된다.

한편, OLTP서버의 CPU 규모산정을 위한 산정항목을 대상으로 분석한 결과 규모추정의 정확도에 가장 큰 영향을 미치는 분당 트랜잭션 수의 산정이 TPC-C의 트랜잭션 정의에 따른 발생건수를 기반으로 산정되지 않으며, 어플리케이션구조보정의 적용기준인 “응답시간” 역시 TPC-C의 응답시간의 충족요건을 고려하지 못하고 있어 개선이 요구되는 상황이다.

우선, 분당트랜잭션수를 살펴보면, TTA의 정보

<표 3> 현행 OLTP서버의 CPU 산정항목의 문제점

구분	산정 항목	내 용	문제점
O1	분당 트랜잭션 수	산정 대상 서버에서의 분당 트랜잭션발생 추정치의 합	TPC-C의 트랜잭션 정의에 입각한 발생건수 산정 필요
O2	기본 tpmC보정	최적의 환경에서 측정된 tpmC 수치를 실제 환경에 맞게 적용하기 위한 보정	
O3	피크타임부하보정	업무가 과중한 시간대에 시스템이 원활하게 운영될 수 있도록 피크타임을 고려한 보정	
O4	데이터베이스크기보정	데이터베이스 테이블의 레코드 건수와 전체 데이터베이스 볼륨을 고려한 보정	
O5	어플리케이션구조보정	어플리케이션의 구조와 요구되는 응답 시간에 따른 성능 차이를 감안한 보정	TPC-C에서 규정하고 있는 응답시간을 고려하여, 세부적인 적용치의 개선이 필요
O6	어플리케이션부하보정	온라인 작업을 수행하는 피크타임에 배치작업 등이 동시에 이루어지는 경우를 감안한 보정	
O7	클러스터보정	클러스터 환경에서 장애발생 시를 대비한 보정	
O8	시스템여유율	예기치 못한 업무의 증가 및 시스템의 안정된 운영을 위한 여유율	

통신단체표준인 “정보시스템 하드웨어 규모산정 지침”의 OLTP서버 규모산정의 기준에 의하면, 규모산정의 기본이 트랜잭션임을 지적하고 있다. 이러한 트랜잭션은 “클라이언트/서버 환경에서 클라이언트에서 서버로의 업무의 요청(Request)과 응답(Response)”으로 정의하고 있으며, 하나의 작업을 하나의 트랜잭션(Transaction)으로 규정하고 있다 [3, 9].

현행 성능 기준이 tpmC임을 감안할 때, 트랜잭션은 TPC-C에서 규정하고 있는 특성을 그대로 상속받아 정의되어야 한다. 따라서 앞서 TPC-C벤치마크에서도 살펴보았지만, TPC-C에서의 트랜잭션 및 tpmC의 개념을 명확히 이해할 필요가 있다. 이러한 트랜잭션과 tpmC의 개념은 다음과 같이 요약할 수 있다.

- TPC-C에서 규정한 트랜잭션은 DB수준의 트랜잭션이 아닌 비즈니스트랜잭션으로 new-order, payment, order-status, deliverly, stock 등 5개로 정의
- 트랜잭션별 처리조건은 최소기준을 정의하고 있는데 payment(43%), order-status(4%), deliverly(4%), stock(4%)이며, new-order는 4개의 트랜잭션 비율에 따라 달라지고 측정대상이 되므로 별도로 정의하지 않음.
- 성능측정 결과의 주요 인정조건으로는 각 트랜잭션이 정해진 응답시간내[new-order(5초 이내), payment(5초 이내), order-status(5초 이내), deliverly(5초 이내), stock(20초 이내)]에 90% 이상 수행 되었을 때, 해당 성능측정을 인정함(단, 응답시간은 서버차원의 응답시간이 아닌 클라이언트 차원의 응답시간을 의미함)
- tpmC는 Keying time(최소 18초), think time(최소평균 12초) 등을 고려하여 위의 인정조건을 만족하였을 때, 처리된 new-order수로 산정함.

위의 이들 조건을 만족하는 경우, TPC-C에 의한 분당 발생 가능한 트랜잭션수는 최대 2개를 넘을 수 없다. 그러나 현행 표준에서는 사용자당 최소 3개에서 최대 6개로 정의하고 있다. 이는 TPC-C에서의 트랜잭션 및 tpmC의 정의로 볼 때, 발생 불가능한 값이다. 따라서 이는 트랜잭션이 규모산정의 기저라는 측면에서 규모산정의 정확성에 매우 큰 영향을 미치게 되며, 이는 결국 과도한 규모산

정(최소 1.5배에서 최대 3배까지)의 결과를 낳게 될 수 있다.

다음은, 어플리케이션구조보정의 적용기준인 “응답시간”이다. 일반적으로 서비스의 응답시간은 사용자의 성능적 만족을 판별하는 중요한 기준이다. 응답시간이 느린 시스템은 사용자를 짜증나게 만들고 나아가 고객을 떠나보내는 중요한 요인으로 작용할 수 있다. 따라서 시스템에 대한 사용자 만족을 고민할 때 응답시간은 필수적인 요소이며, 중요한 항목이라고 할 수 있다. 따라서 산정기준에서도 어플리케이션부하 보정을 고려할 때, 이러한 응답시간에 따라 보정치를 달리하고 있다.

<표 4> 현행 어플리케이션 구조보정 적용방법

응답시간 구조	3초 이내	5초 이내	5초 이상
1-계층 구조	100%	80%	60%
2-계층 구조	80%	60%	40%
3-계층 구조	40%	30%	10%

한편, 현행의 어플리케이션 구조보정은 구축하고자하는 시스템의 계층구조와 응답시간을 고려하여 적용하고 있다. 이러한 계층구조는 각 서버의 CPU가 산정되면 구축하고자하는 시스템의 계층구조를 감안하여 참조모델(1계층, 2계층, 3계층)별 적용값을 달리하고 있어, 계층구조를 고려하고 있다. 따라서 CPU 산정항목에서 계층구조를 감안할 경우, 이종의 보정치를 주게 됨으로 이를 고려하지 않는 것이 바람직하다. 또한 응답시간의 경우, 응답시간이 5초 이상에 대해서도 보정치를 부여하고 있으나 TPC-C에서 규정에 의해 stock-level 트랜잭션을 제외한 4가지 트랜잭션이 5초 이내에 수행되어야 함을 감안할 경우, 현행과 같이 5초 이상에 대해서 보정치를 주는 것은 불합리하다.

### 3.3 개선된 규모산정 모델

OLTP서버의 CPU에 대한 규모는 <표 3>에서

와 같이 분당트랜잭션수, 기본 tpmC 보정, 시스템 여유율 등에 8개 산정항목 중 분당트랜잭션수와 어플리케이션구조보정의 두 가지 산정항목에 대한 개선된 세부적용 방법을 기술하였다.

### 3.3.1 분당 트랜잭션 수

OLTP 환경하에서 시스템 규모산정의 정확성을 위해서는 시작점이 중요하다. 그 시작점으로 기본이 되는 것이 바로 트랜잭션이다. 이는 클라이언트/서버 환경에서 클라이언트에서 서버로의 업무의 요청(Request)과 응답(Response)으로 정의할 수 있다. 즉 하나의 작업을 하나의 트랜잭션(Transaction)으로 칭한다.

클라이언트/서버 환경하에서는 일반적인 응용구조가 이와 같이 트랜잭션 단위로 업무가 발생하므로 한 어플리케이션에서 과연 몇 번의 트랜잭션이 발생하는가를 추정하는 것이 OLTP 환경에서 시스템 규모산정의 기본이 된다. 이러한 분당 트랜잭션 수를 구하는 방법은 일반적으로 기존시스템의 트랜잭션을 조사하는 방법, 동시사용자 수를 이용하는 방법, 클라이언트 수를 이용하는 방법 등 3가지 방법이 있다.

〈표 5〉 트랜잭션 수 적용방법

구 분	적용 값의 범위	일반 값
동시사용자수	총사용자의 10%~30%	20%
사용자당 트랜잭션 처리수	업무의 유형 및 복잡도를 감안하여 2~3 값을 적용	2

분당 트랜잭션 수 = 동시사용자 수 × 사용자당 트랜잭션 처리수

신규시스템 도입과 같이 기존에 조사되어진 트랜잭션이 존재하지 않는 경우(즉 시스템에 대한 예상치와 향후 개발될 어플리케이션에 대한 구체적인 내용을 산정하기 어려운 경우)에는 총사용자 수를 예상하여 동시사용자 수를 구하고, 예상되는 업무유형과 특성을 감안하여 동시사용자 한 사람이 분당 발생 가능한 예상 트랜잭션 수(사용자당 트

랜잭션 처리수)를 추정하여, 이를 바탕으로 최종적인 분당 트랜잭션 수를 구할 수 있다.

### 3.3.2 어플리케이션 구조 보정

어플리케이션 구조 보정은 어플리케이션 응답시간에 따른 성능 차이를 감안하기 위한 보정치이다. 즉, 어플리케이션 사용자에게 원하는 적정 서비스 응답시간을 보장하기 위해서는 서버의 CPU 자원의 성능도 달라져야 한다.

개선된 어플리케이션 구조 보정을 위한 적용값은 <표 6>에서와 같다. 이는 기존 방식에서의 문제점으로 지적된 tpmc의 응답시간 규정에 위배되는 5초 이상의 응답시간에 대해서 추가적인 보정치를 부여하지 않았다. 또한, 이중적인 계층구조에 대한 반영을 해소하기 위하여 기존 어플리케이션 구조 보정에서 사용하였던 계층구조 및 응답시간에 따른 적용값 대신에 응답시간에 대해서만 적용값을 부여하도록 보완하였다.

〈표 6〉 어플리케이션 구조보정 적용방법

응답시간	1초	2초	3초	4초
적용 값	50%	35%	20%	10%

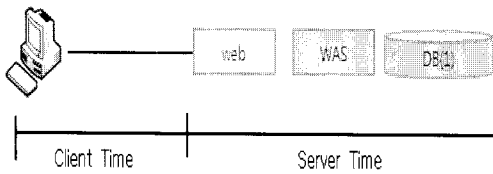
여기에서의 응답시간은 서버의 응답시간을 의미하는 것이 아니며, 사용자의 서비스 응답시간을 말한다. 이러한 서비스의 응답시간은 사용자의 성능적인 만족을 판별하는 중요한 기준이다. 따라서 서비스 응답시간에 대한 명확한 개념적정의가 필요하다.

응답시간이 느린 시스템은 사용자의 만족을 현저하게 떨어뜨림으로써 회사의 고객을 떠나보내는 중요한 요인으로 작용할 수 있다. 따라서 시스템에 대한 사용자 만족을 고민할 때 응답시간은 필수적인 요소이며, 중요한 항목이라고 할 수 있다.

응답시간이라 함은 사용자가 업무 요청내용을 시스템에서 처리후 처리 내용의 결과를 출력하는 전체 시간을 의미한다. [그림 2]에서와 같이 응답시간을 크게 구분하면 서버시간과 클라이언트시간으로 구분되어 질수 있는데 서버시간은 WEB 서



버 수행시간, WAS서버 수행시간, DB 서버 수행 시간, 서버간 네트워크 시간을 합한 전체 소요시간을 의미하고, 클라이언트시간은 데이터 네트워크 전송시간, 클라이언트 수행시간의 전체 소요시간을 의미한다.



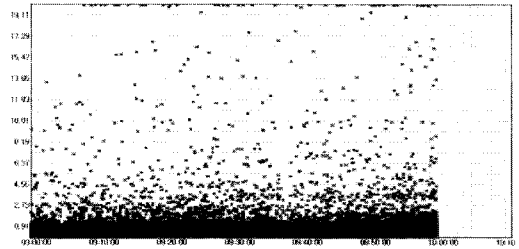
[그림 2] 응답시간 구분

위 기준으로 응답시간이 3초라함은 서버시간과 클라이언트시간의 합을 기준으로 하는데 이때 서버시간이 3초가 소요되는 경우 사용자의 응답시간은 3초에 클라이언트시간을 더한 시간이 소요된다. 그러나 사용자의 입장에서는 두 구간의 응답시간을 구분할 수 없으며, 전체 응답시간 즉 사용자의 체감 응답시간이 3초로 인식하는 경우가 일반적이다. 따라서 시스템상의 성능시험 시 응답시간의 목표가 3초의 경우에는 클라이언트 시간을 제외한 응답시간임으로 사용자가 3초의 응답시간을 만족하기 위해서는 서버시간의 경우에는 3초미만의 응답시간 기준을 만족해야 한다.

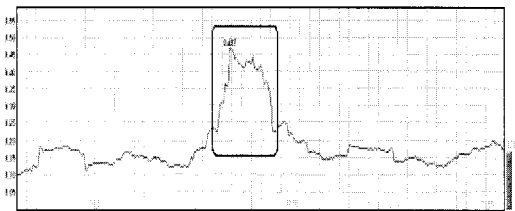
성능테스트에서는 최상의 환경으로 네트워크 병목이나, 사용자 컴퓨터의 병목은 고려되어지지 않은 상태에서 테스트를 수행하고, 산출된 결과를 응답시간으로 정의하고 있다. 단 성능 테스트는 실제 운영 환경과 동일한 환경으로 수행되어지지 않고, 처리량도 실제 운영시보다 높은 처리량을 기준으로 성능테스트를 수행함으로 이에 대한 응답시간 보정은 감안해야 한다.

시스템에서 응답시간은 각각의 사용자가 서로 다른 응답시간을 나타내는데, 그래서 응답시간은 개별 서비스에 대해서 하나의 점으로 모니터링 되어진다. 시스템 관리차원에서는 이 각각의 응답시간의 분포의 평균을 기준으로 응답시간을 모니

터링 되고 있다. 따라서 응답시간의 기준으로 [그림 3]의 결과값을 평균 낸 [그림 4]의 결과이며, 이 중 응답시간은 피크(Peak)시 평균 응답시간이 3초의 기준을 만족하는 경우로 정의 할 수 있다.



[그림 3] 응답시간 개별 분포



[그림 4] 일평균 응답시간 분포

#### 4. 결 론

하드웨어 규모산정은 도입되는 시스템을 운영할 업무의 성격과 업무의 예상증가율, 구축기술의 특성들을 전체적으로 고려하여 산정해야 하므로 규모산정의 적정성을 직관적으로 판단하는 일은 매우 어려운 일이다. 따라서 이러한 어려움에도 불구하고 공공부문 정보화자원의 효율적 운영을 통한 정보화 투자효과의 극대화를 위해서는 정확한 규모산정이 필요하다.

본 연구는 이러한 정보시스템 하드웨어 규모산정 방식 중 OLTP서버 산정모델에 대한 개선의 일환으로써 보다 정확한 규모산정을 위하여 기존지침이 가지고 있는 문제점을 파악하고 개선된 모델을 제시하는데 그 목적이 있다. 이를 위해 정보시스템 하드웨어 규모산정 지침과 기존 규모산정 연구를 기초로 하여 문제점에 대한 분석과 개선 모

델을 수립하고 관련 전문가의 검토(탐색적 사례연구 및 전문가 검토) 과정을 통해서 최종적인 산정 방식을 제안하였다.

향후 본 연구를 반영하여 정보통신단체표준(TTA Standard)인 “정보시스템 하드웨어 규모산정 지침”의 개정을 통해서 지침의 신뢰성과 규모산정의 정확성을 높이는 것이 필요하다.

## 참 고 문 헌

- [1] 나중희, 최광돈, “계산방식에 기반한 정보시스템 도입 규모 추정”, 『한국SI학회지』, 제5권, 제1호(2005), pp.9-25.
- [2] 나중희, 최광돈, “정보시스템 용량산정방식에 관한 탐색적 연구 : 공공부문 H/W 규모산정을 중심으로”, 『한국SI학회지』, 제3권, 제2호(2004), pp.9-24.
- [3] 박성환, 장주연, 서영주, 박원주, “플래쉬 변환 계층에 대한 TPC-C 벤치마크를 통한 성능분석”, 『한국정보과학회논문지』, 제14권, 제2호(2008), pp.201-205.
- [4] 정해용, 나중희, 최광돈, “공공부문 정보시스템의 하드웨어 용량산정 방식 설정을 위한 실증적 연구”, 『정보화정책』, 제12권, 제3호(2005), pp.54-72.
- [5] 한국전산원, “H/W 용량산정 관련 연구”, 2002.
- [6] 한국전산원, “정보시스템 용량산정기술 및 프레임워크 연구”, 2003.
- [7] 한국전산원, “정보시스템 규모별 용량산정 기준 연구”, 2004.
- [8] 한국정보통신기술협회, “정보시스템 하드웨어 규모산정지침”, 2008.
- [9] TPC, “TPC Benchmark C Standard Specification Revision 5.9”, 2007.
- [10] [www.spec.org](http://www.spec.org).
- [11] [www.tpc.org](http://www.tpc.org).

## ◆ 저 자 소 개 ◆

**나 종 회 (jhra@gwangju.ac.kr)**

성균관대학교 공과대학 정보공학과에서 학사, 석사, 박사를 취득하였다. 한국전산원 감리본부에서 주임연구원으로 근무하였으며, 현재 광주대학교 e-비즈니스학과 부교수로 재직 중이다. 한국정보처리학회지, 정보화정책, Information Systems Review 등의 국내학술지에 논문을 게재한 바 있으며 관심분야는 e-비즈니스, 정보화사업평가 및 감리, 정보시스템 성능 등이다.

**최 광 돈 (kdchoi@hansei.ac.kr)**

한국외국어대학교 경영정보대학원에서 석사, 광운대학교에서 박사를 취득하였다. 한국생산성본부 MIS실 실장으로 근무하였으며, 현재 한세대학교 경영학부 e-비즈니스전공 부교수로 재직 중이다. 한국경영과학회지, 정보화정책, 한국디지털정책학회지 등에 논문을 게재하였으며 관심분야는 ERP운영, 성과관리, 정보화사업평가 및 감리, 정보시스템 성능 등이다.