

ATM 교환기의 시험평가 방법 (Testing Evaluation Method of ATM Exchange)

정승국* 노성기** 정택원***
(S.-K. Cheong, S.-K. Noh, T.-W. Jeong)

본 고는 ATM 교환시스템 연구개발 단계의 품질보증 활동의 일환인 시스템 시험평가에 적용할 목적으로 연구된 것으로서, ATM 시스템 시험평가에 필요한 기초요소, 에러의 범위, 시험평가의 개념 등을 체계화하고 정의하여 ATM 시험평가에 대한 기반을 구축하였으며, 시험평가에 대한 구체적인 방법을 제시하였다. 이 체계화된 시험평가 방법의 적용은 연구개발 단계의 ATM 교환시스템에 대한 지속적인 품질 향상을 이루며, 안정성과 신뢰성을 갖는 고품질 ATM 교환시스템이 확보되도록 할 것이다.

I. 서 론

교환(exchange)이라는 개념은 1980년대 이전에는 주로 전화와 전화를 연결하는 음성 스위칭에서만 사용되었으나, 1980년대 중반 이후부터 컴퓨터의 급속한 발달과 함께 다양한 통신 서비스가 요구되었고, 이에 따라 통신에서의 교환이라는 개념도 음성뿐만 아니라, 컴퓨터간의 통신 및 화상데이터 전송 등과 같은 비음성 통신을 통합하는 형태로 발전하였다.

초기에는 협대역 종합정보통신망(narrowband ISDN; N-ISDN)의 개념 하에 음성과 비음성 통신을 통합하는 형태로 연구되었으나 동화상 데이터 전송과 같은 고속의 데이터 통신을 요구하는 다양한 서비스가 대두됨에 따라 음성 교환 위주의 정보 교환에서 사용되었던 64kbps 스위칭 속도로는 그러한 서비스의 요구를 만족시킬 수 없었으므로 수십 Mbps의 고속 데이터 교환이 가능한 새로운 스위칭시스템이 필요하게 되었다. 이에 따라 ITU-T에서는 광대역 통신용으로 비동기 전달 모드(asynchronous transfer mode; ATM)를 채택하였으며, 세계 각국에서는 ATM 교환기의 구현을 목표로 활발한 연구가 진행되고 있다. 우리나라에서도 이 ATM 스위칭 기술에 대한 연구 개발

* 품질보증연구실 선임기술원

** 품질보증연구실 연구원

*** 품질보증연구실 실장

이 한국전자통신연구소를 중심으로 산업계 및 학계에서 활발히 진행되고 있다.

대규모 시스템의 연구 개발에 있어서, 연구 개발품의 디자인(design)과 실현(implementation)을 검증하기 위해서는 시험평가 기술이 필수적인 요소이다. 시험평가는 하드웨어와 소프트웨어로 구성되는 전반적인 시스템을 대상으로 이루어지는 품질 확인 절차로서 연구개발 단계에서는 필연적으로 발생하는 행위(activity)이며, 이것은 시스템 연구개발의 성공과 실패를 좌우한다[1]. 그러나, 기존의 연구개발 단계에서는 시험평가에 대한 공식화된(formulated) 체계적 방법이 거의 없었다. 따라서, 앞으로 진행될 ATM 연구개발 단계에 대한 시험평가 방법을 체계화하여 시험 평가의 기반을 구축하는 것은 초기 개발 단계에 있는 현재 상황에서는 매우 중요한 일이다. 또한, 이러한 체계화된 내용을 토대로 연구개발품에 대한 검증이 이루어진다면, 연구개발품의 지속적인 품질 향상이 이루어질 수 있으며, 개발단계의 시스템에 대한 안정성과 신뢰성을 확보할 수 있을 것이다.

이의 체계 정립을 위하여 II장에서는 시험평가 방법의 기초 요소들을 선정하고, III장에서는 시험평가 방법에 대해 논의한다. IV장에서는 III장에서 제안한 시험평가 방법을 토대로 시험평가에 관한 체계를 확립하고, 마지막으로 V장에서 결론을 맺는다.

II. 시험평가 방법의 기초 요소

시험평가는 하드웨어와 소프트웨어로 구성되는 대규모 시스템 개발에 있어서 연구개발품의 품질을 확인 및 검증(verification and validation)하는 행위로

정의된다. 이를 위해서는 개발 제품에 대한 시험(testing)이 선행되어야 하며, 시험수행(testing execution)을 통해 표출된 결과(하드웨어 및 소프트웨어 에러, 시험자 요구사항 등)를 분석(analysis)하고 평가(evaluation)하는 일이 수반되어야 한다[2]. 그러므로 시험에 관련된 기초 요소들이 정의되어야 하는데, 기존에는 이러한 내용들이 구체화되지 못했고, 개발지침서 등의 일부에 기술된 정도였다. 따라서, 개발단계의 시험평가에 대한 기초 요소를 <표 1>과 같이 정의하였다

<표 1> 시험평가 기초 요소

분야	기 초 요 소
시험 행위	<ul style="list-style-type: none"> o. 시험 계획 o. 시험 절차 설계 <ul style="list-style-type: none"> - 시험 항목 설정 - 시험 절차서 작성 o. 시험 수행 o. 에러 분류 o. 시험 결과 분석
시험 기술	<ul style="list-style-type: none"> o. 기능시험 기술 <ul style="list-style-type: none"> - Hardware Oriented Test - Software Function Test o. 시스템 시험 기술 <ul style="list-style-type: none"> - 성능 시험 - 종합 시험 - 반복 시험
시험 환경 구축	<ul style="list-style-type: none"> o. 시험용 교환기 확보 및 관리 o. 망 연동 환경 o. 기능 시험 환경 o. 성능 시험 환경
시험 내역 관리	<ul style="list-style-type: none"> o. 시험 결과 종합 및 분석 o. 시험 종류별 내역 o. 시험 항목별 내역 o. 개발 단계별 내역 o. 소프트웨어 버전별 내역

이러한 평가요소들 중 시험기술, 시험환경구축, 시험내역(test history) 관리[3] 등은 모두 시험행위(activity)를 위한 주변 기능적 요소이므로 무엇보다도 시험 행위에 대한 정확한 정의가 필요하다. 시험행위는 개발 단계마다 정의되는 가변적인 것으로서 가급적이면 타이트하고 정교한 내용으로 작성되고 계획되어야 한다.

Ⅲ. 시험평가 방법

시험평가의 목적은 시험에서 발생한 데이터를 정확히 분석하여 제시함으로써 다음 개발 단계의 지표(milestone)가 되도록 하는 것이므로 가급적이면 객관적인 시험결과 분석을 토대로 정확한 시험평가가 이루어져야 한다. 그러나, 기존의 시험평가 방법은 시험항목 하나 하나의 중요성이나 발생한 에러(error) 개수에 관계없이 묵시적으로 시험항목의 합격률이 연구개발 단계의 시스템에 대한 품질수준인 것으로 생각하였다. 이러한 평가방법은 엄밀히 이야기하면 시험평가라기보다는 단지 시험항목의 합격률을 백분율로만 표현한 정도였다. 예를 들어, 시스템 시험항목이 100개이고, 그중 70개의 항목이 양호하며, 30개의 항목이 불량이라면, 70%의 합격률로 표현하고, 시스템에 대한 평가치가 마치 70% 정도인 것처럼 생각하였다. 이 예에서, 30개의 불량 항목이 가지는 여러 가지 에러에 대한 내용들은 시스템에 미치는 영향에 따라 각기 다르고, 각각의 항목이 가지고 있는 에러들의 개수가 서로 다르기 때문에 70%의 합격률이라는 표현은 적절하지 못하다. 따라서, 시험을 통한 시스템평가라는 범위에서 생각하면 시스템평가의 척도를 시험대상항목수의 합격률로만 삼았던 기

존 생각은 매우 불합리한 것이었으며, 개발일정 지연, 시스템최적화(optimization) 실패 등의 결과를 야기시켰다.

이와 같은 기존의 묵시적인 시스템평가방법은 누구나 잘못되었다고 생각하면서도, 보다 정확한 시스템시험평가에 대한 방법을 제시하지는 못했다. 그러므로, 현재 진행되고 있는 ATM 시스템개발에 있어서는 개발 단계별로 정확한 시험평가를 수행하여 개발 초기에 계획된 개발일정에 따라 연구개발품의 품질이 향상되도록 하고, 시스템의 최적화를 이루도록 하기 위해서는 시험평가에 대한 방법이 정의되고 체계화되어야 한다. 즉, ATM 연구개발 단계에서의 시스템에 대한 평가는 기존의 관례적인 시험평가방법(시험항목수의 합격률)이 아닌 최대한 구체적이고 정량화된 단위로 표현되고 평가되어야 한다. 따라서, 본 연구에서는 ATM 연구개발 단계의 시험평가 방법에 대해 다음과 같이 2가지 방법을 제안한다.

방법 1) 에러의 임계치 평가(threshold evaluation) :

개발 단계의 에러들의 수가 미리 정의한 임계값을 기준으로 평가하는 방법

방법 2) 에러의 상호 관계와 중요도에 의한 평가:

시험을 통하여 표출된 에러들의 수 및 시험 항목들이 가지는 상호 연관 관계(correlation) 및 중요도(weighting factor)를 세밀히 기술하고 분석하여 종합 평가하는 방법

이 방법들은 기존의 시험항목 합격률이라는 개념과는 달리 시험을 통해 표출된 에러들에 의해 시스템에 대한 평가가 이루어지도록 한 것이다. 따라서, 이 방법들을 실현하기 위해서는 시스템 시험에서 발견된 에러들에 대한 명확한 정의와 분류가 선행되어

야 한다.

1. 에러의 범위

일반적으로 에러(error)는 시스템(하드웨어 및 소프트웨어)에 내재되어 있는 오류들(bugs)로 정의되며, 시험(testing)은 이러한 오류를 찾아내는 활동으로 표현된다[4, 5]. 이러한 각각의 에러들은 시스템에 미치는 영향이나 어떤 block 혹은 function에 작용하는 상호 관련성에 따라 그 에러가 가지는 중요도가 서로 다르므로 이러한 에러가 가지는 중요성의 차이에 따라 에러의 범위가 결정되어야 한다. 그러나, 지금까지(최소한 N-ISDN 연구 개발 단계 까지)는 이러한 에러에 대해 구체적인 분류나 명확한 정의가 없었으므로, 본 연구에서 제시한 시험평가방법을 구체화하기 위해서 에러의 범주(categories)를 <표 2>와 같이 분류한다.

시스템 에러에 있어서 성능 에러란 프로세서 메모

<표 2> 에러의 범위

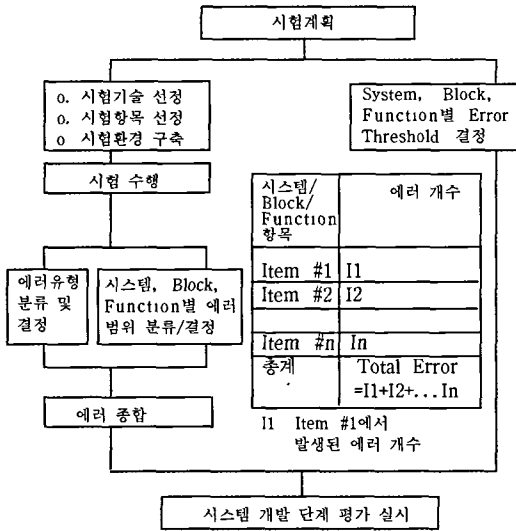
구분	종류	내용
시스템에러	- 성능 에러 - 타이밍 에러	하드웨어와 소프트웨어의 각 기능들이 가지는 에러를 제외한 시스템 전반에 걸쳐 복합적인 형태로 나타나는 에러
하드웨어 에러	- 설계 에러 - 펌웨어 에러 - 하드웨어 고장으로 인한 에러	소프트웨어와 독립적으로 발생하는 에러로서 하드웨어 설계 오류 및 펌웨어의 로직 에러에 의해 발생하는 에러
소프트웨어 에러	- 설계 에러 - OS 에러 - DB 에러 - 응용소프트웨어 에러	하드웨어에 의존적이면서 기능 구현 차원에서 OS, DB, 응용 프로그램들이 가지는 모든 종류의 소프트웨어 에러

리 영역에서 발생하는 overflow 형태나 프로세서 능력밖의 어떤 계산상의 요구(CPU 이용률, 메모리 할당 등) 행위에 의해 발생하는 에러들이며, 타이밍 에러는 교환기와 같은 실시간 시스템에서 가장 중요한 에러로서 한번 발생한 에러의 형태가 재발생되기 어려운 형태의 에러이다. 또한, 소프트웨어 에러중 설계에러는 문서에러(documentation error)를 포함하는데, 이 경우는 개발단계 정의문서(사용자요구사항, 기술문서 등)의 오류에 의해 잘못 설계된 에러들을 포함한다.

2. 시험평가 방법의 개념

본 연구에서 제안한 두가지 시험평가 방법에 대한 개념을 (그림 1)과 (그림 2)에 각각 나타내었다. 방법 1에 대한 구체적인 내용은 시험결과에 따라 다양한 에러들이 각 시험항목에 따라 나열되고 그 나열된 에러들의 수가 종합됨으로써, 시스템 전체의 내용이 평가되도록 하는 방식이다. 이 경우, 에러의 임계치는 시험계획 단계에서 미리 정의되어야 하고, 임계치를 기준으로 상대평가를 수행한다. 평가는 시스템의 block이나 function에 따라 정의될 수도 있고, 시스템 전체로 정의될 수도 있다. 따라서, 이 방법은 시험항목의 양부에 관계되지 않고, 시스템의 block 혹은 function 별로 평가가 가능하다. 그러나, 어느 특징기능 혹은 특정항목 하나가 가지는 에러의 수가 많으면 많을수록 그 기능이나 항목이 차지하는 비중이 너무 높아지기 때문에 분석 결과에 대한 정확성이 떨어지는 단점이 발생할 수 있다.

방법 2는 방법 1의 단점이 보완된 형태이다. 상에서 설명한 바와 같이 평가 방법 1은 발생한 에러의

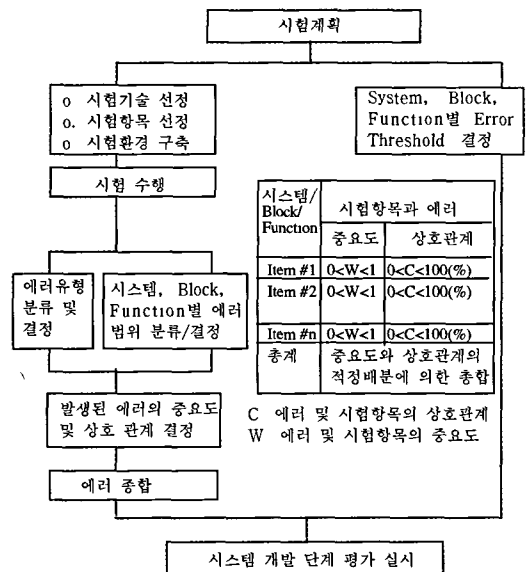


(그림 1) 시험 평가 방법 1의 개념

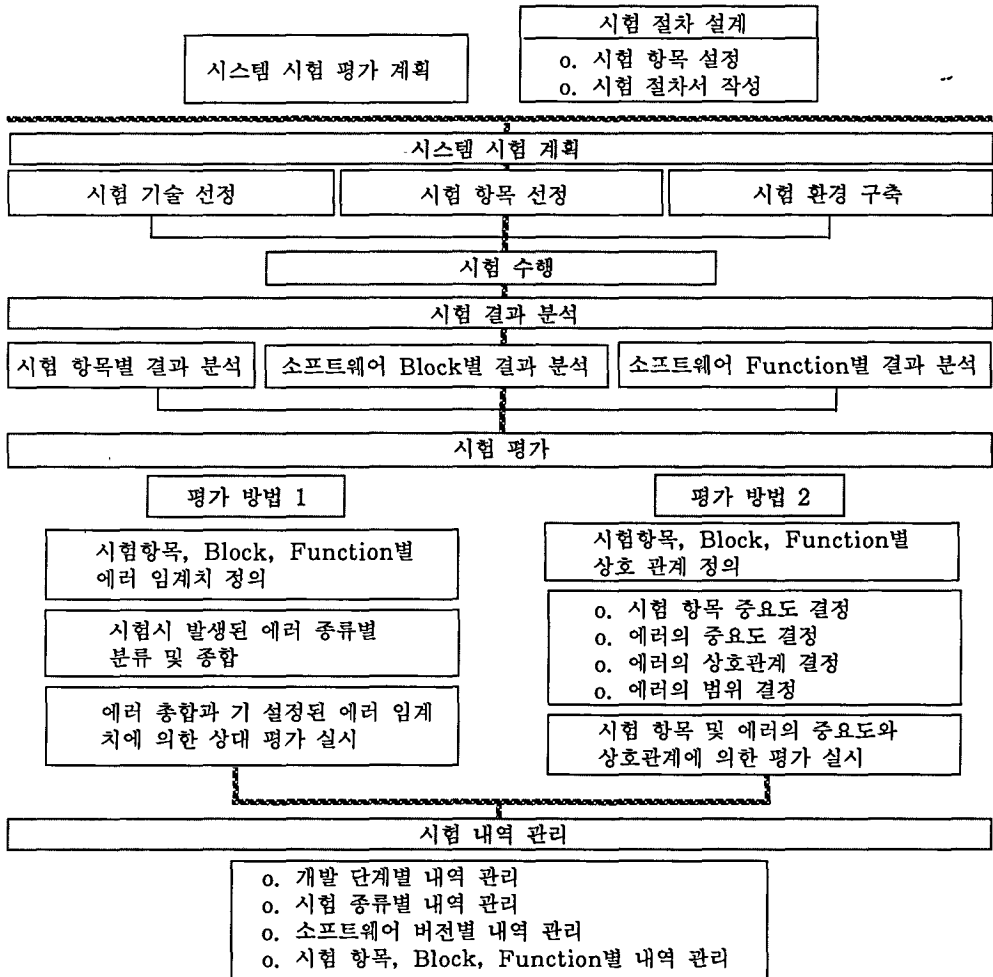
개수 하나 하나가 모두 동일한 가중치로 작용하므로 하나의 기능 혹은 항목에 대해 너무 많은 에러가 발생하면 그 기능 혹은 항목이 차지하는 비중이 너무 커져 최종 시스템의 전반적인 평가에 대한 정확성이 떨어진다. 반면 방법 2는 (그림 2)에서 보는 바와 같이 각 에러(error)에 대해 각 시험항목(test item) 및 타 에러들과의 상호관계(correlation)와 중요도(weighting factor)를 고려함으로써 하나의 기능 혹은 항목에 대해 아무리 많은 에러가 발생되어도 그 기능 혹은 항목이 갖는 비중이 알맞게 조정되도록 한 것으로, 보다 정확한 시험평가 결과를 확보할 수 있다. 이 방법은 상호 관련성과 중요도에 의해 평가의 정확성이 좌우되므로 객관성 확보를 위한 노력이 절대적으로 필요하다. 객관성 확보를 위해서는 하드웨어 및 소프트웨어 개발자와 시스템 기술자간의 공동 기술 경험을 토대로 수행되어야 하는데, 이러한 행위는 개발 과정의 시험결과에 따라 다른 형태로 정의되기

때문에 개발 초기에 어느 일정한 값으로 정의될 수는 없으며, 개발단계의 시험행위가 발생될 때마다 시험항목 및 에러에 대한 상호관계와 중요도를 매번 재정의하여야 하는 번거로움이 발생된다.

두 방법의 장단점을 살펴보면, 첫번째 방법은 구현이 쉬운 반면에 분석 결과에 대한 정확성이 약간 결여되는 단점이 있고, 두번째 방법은 중요도와 상호관계의 정의가 잘못하면 너무 주관적이 되기 쉬울 뿐만 아니라 중요도와 상호관계의 적용을 어떻게 하느냐에 따라 분석 결과가 달라질 수 있으므로 평가 방법의 구체적인 실현이 조금 어려운 단점이 존재한다. ATM 교환기의 개발 단계 시스템에 대한 시험평가에 있어서, 보다 정확한 시험결과 분석을 통한 품질 향상을 이루기 위해서는 평가 방법의 실현이 조금 어렵기는 하지만 두번째 방법을 채택하는 것이 바람직하며, 구체적인 시험 분석 기법 등은 추후에



(그림 2) 시험 평가 방법 2의 개념



(그림 3) 시험평가 체계

연구되어야 한다.

IV. 시험평가 체계

II장에서 정의한 시험평가의 기초 요소와 III장에서 제시한 시험평가 방법을 ATM 연구 개발 단계에 적용하기 위해서는 종합적인 시험평가 체계가 확립

되어야 하며, 이의 체계는 (그림 3)에 제시하였다. 연구개발 초기 단계에서는 시험평가의 장기적 계획과 시험을 위한 시험항목 설정, 시험절차서 작성 등이 선행되어야 하는데, 이 행위는 개발 초기에서만 이루어진다. 따라서, 시험절차 설계 단계는 목표하는 시스템의 기능이 모두 포함되는 형태로 설계되어야 하며, 이것은 모든 개발 단계의 기능이 포함되어야

한다.

시스템시험계획단계부터는 개발단계에 적용되는 시험계획에 따라 순환 형태로 계속적으로 반복된다.

V. 결 론

본 고에서는 ATM 개발 단계의 시험 평가에 대한 기초 요소를 정의하고 그에 따른 체계를 확립하였으며, 시험 평가에 관한 두가지 방법을 제시하였고, 시험평가시 가장 기본적인 요소(element)가 되는 시험 결과의 에러에 대한 범위를 결정하고 정의하였다. 여기서 제안된 시험평가 방법은 현재 연구가 진행되고 있는 ATM 교환 시스템 연구 개발 단계에 적용하여 연구 개발품(ATM교환기)의 지속적인 품질 향상을 유도하고, 개발단계의 시스템에 대한 안정성과 신뢰성을 확보할 수 있도록 할 것이다.

이 평가 방법의 실현을 위해서는 III장 2절에서 논의한 바와 같이 시험 평가에 대한 구체적인 분석방법이 여러가지 형태로 구현될 수 있도록 기초 요소, 에러유형, 중요도, 상호관계 등이 개발 단계별로 정확히 정의되도록 하여야 하며, 그들의 구체적인 값은 종합적인 데이터를 이용한 튜닝 작업이 필요하므

로 추후에 연구되어야 할 대상이다. 시험평가는 별도의 시험 결과 분석 기법에 의하여 자동 검증(validation) 되도록 하여야 하고, 그에 따라 개발 단계의 완성도를 백분율로 관리할 수 있도록 하여야 할 것이다.

참 고 문 헌

- [1] Martin L. Katcher, "Integration and System Testing: a Methodology," *ISS Session 14 C Paper 1*, pp. 1-5, 7-11, May 1984.
- [2] L.J.Osterweil, "A Strategy for Integrating Program Testing and Analysis," *ed. by B Chandrasekaran and S.Radicchi, Computer Program Testing*, N.H., pp 187-230, 1981
- [3] M Cerani, A Cicu, M.Maiocchi, "A Methodology for Accurate Software Test Specification and Auditing," B Chandrasekaran and S Radicchi, *Computer Program Testing*, N.H., pp 301-325, 1981.
- [4] L.J.Osterweil, L.D.Fosdick, R.N.Taylor, "Error and Anomaly Diagnosis through Data Flow Analysis," B.Chandrasekaran and S Radicchi, *Computer Program Testing*, N.H., pp 35-63, 1981.
- [5] WE Howden, "Errors, Design Properties and Functional Program Tests," B Chandrasekaran and S.Radicchi, *Computer Program Testing*, N.H., pp115-127, 1981.