

# 실시간 객체 모형을 이용한 상호운용성 시험 접근 방법

최진영<sup>†</sup> · 민병준<sup>††</sup> · 김문희<sup>†††</sup>

## 요약

다수 공급자에 의해 제공되는 여러 구현들간의 상호운용성 시험은 많은 비용을 초래한 뿐 아니라 실제 상황에서 때로는 그 자체가 불가능하다. 적합성 시험이나 적합성 시험을 확장한 방법들 만으로는 효과적으로 상호운용성을 보장할 수 없다. 본 논문에서는 실시간 시스템인 통신망의 구성 요소들을 매우 훌륭하게 추상화할 수 있는 실시간 객체 모형을 기반으로 한 프로토콜 상호운용성 시험 방법을 제시한다. 이 방법에서는 RTO.k 객체 모형을 이용하여 IUT(Implementation Under Test)와 시험 환경을 추상화하고 그 결과 얻어진 객체들을 실행 환경인 DREAM 커널 상에 지역 모니터와 전역 모니터를 두어 수행시킨다. 지역 모니터는 관찰과 제어를 주요 임무로 하고 전역 모니터는 시험 과정을 주관한다. 시험 방법의 검증은 위하여 B-ISDN UNI Q.2931의 예를 들어 호 연결 제어 프로토콜 상호운용성 시험을 실험하였다. 이 방법으로 높은 결함 발견율을 얻을 수 있다.

## An Approach to Interoperability Test using an RTO Model

Jin Young Choi<sup>†</sup> · Byoung-Joon Min<sup>††</sup> · Moon Hae Kim<sup>†††</sup>

## ABSTRACT

Testing interoperability among various implementations provided by multi-vendors is not only costly but also impossible in many real situations. The conformance testing and its extended testing methodologies are not sufficient to effectively guarantee the interoperability. This paper proposes an approach to the protocol interoperability test based on a real-time object model which successfully abstracts components of real-time systems such as communication networks. In this approach, IUT (Implementation Under Test) and the testing environment are represented by means of the RTO.k model and the resultant objects are executed on the DREAM kernel with local monitors and a global monitor. The local monitors observe and control the events, and the global monitor manages the testing procedure. For the validation of the approach, we experimented an interoperability testing of the call control protocol of B-ISDN UNI Q.2931 as an example. It indicated that high fault coverage could be obtained with this approach.

## 1. 서론

복잡하고 다양한 정보통신 서비스를 위한 여러 제

품들이 다수 공급자에 의해 개발, 보급되고 있다. 이러한 제품들은 상용화되기까지 여러 시험을 거치게 되는데, 특히, 정보통신 서비스의 근간을 이루는 통신 프로토콜의 경우, 개발 시험(development testing), 적합성 시험(conformance testing), 상호운용성 시험(interoperability testing), 강건도 시험(robustness testing), 그리고, 성능 시험(performance testing) 등이 필요하

† 정희원:한국전자통신연구소 연구원

†† 정희원:시립인천대학교 전자계산학과

††† 정희원:건국대학교 컴퓨터공학과

논문접수:1997년 3월 5일, 심사완료:1997년 4월 15일

다. 각 시험은 고유한 적용 범위와 목적을 가지며 시험 대상 구현물(IUT: Implementation Under Test)에 요구되는 신뢰성의 정도에 따라 다르게 적용될 수 있다. 프로토콜 적합성 시험에 대해서는 시험 방법, 시험 구성 및 시험 규격의 기술 등에 관한 많은 연구가 수행되었고, 그 결과들이 국제 기구에 의해 표준화되었다 [2]. 그러나 상호운용성 시험에 대해서는 구체적인 시험 방법론 및 시험 구성에 대한 표준화가 이루어지지 않은 상태이며 제한적으로 프로토콜 일부 계층에 대한 시험 규격 개발이 진행되고 있다 [1].

상호운용성 시험은 실제 상황과 같은 조건에서 구현 제품들이 상호 작용하는 가를 관찰하는 것이다. 표준에 맞게 구현되었는가를 시험하는 적합성 시험에 통과한 구현 제품들에 대해서 주로 시험이 이루어진다. 적합성 시험만으로는 상호운용성을 보장할 수 없는데 그 이유는 다음과 같다 [10].

- 표준 자체를 완벽하게 기술하는 데에 한계가 있어서 규격에 대한 오해의 소지가 있다.
- 표준 프로파일(profile)의 선택 사항들에 의해서 구현들 간에 다른 기능들이 존재할 수 있다.
- 기술적, 경제적 한계로 인하여 적합성 시험을 완벽하게 할 수 없다.
- 적합성 시험 만으로는 성능이나 신뢰도에 대한 평가를 내릴 수 없다.

기존의 상호운용성 시험은 관찰과 제어를 가할 수 있는 정도에 따라 모니터가 없는 경우, 수동 모니터에 의한 경우, 능동 모니터에 의한 경우로 분류될 수 있다 [4]. 능동 모니터를 사용할 경우는 두 구현들 간의 데이터 통신 내용을 관찰할 수 있을 뿐 아니라 오류를 발생시켜서 그에 대한 반응을 살펴볼 수도 있다. 이렇게 함으로써 높은 결함 검출율을 얻을 수 있다. 그러나 기존의 방법들은 하나의 통신망(또는 시스템)을 구성하는 여러 요소들을 통합 시험하지 못하고 통신망을 부분적으로 나누어서 상호운용성을 확인하는 정도에 머무르고 있다. 예를 들어, B-ISDN (Broadband Integrated Service Digital Network)의 UNI(User-Network Interface) 프로토콜에 준하는 두 개의 서로 다른 공급 업체에서 제공된 사용자 단말기의 상호운용성을 시험한다고 하자. 기존의 대부분의 시험 방법에서는 두 단말기 사이에 시험기를 위치시키고 한 단말기에서 호를 발생시킨 후 그 동작의 결

과를 시험기가 다른 단말기에 전달하여 제대로 동작하는 가를 확인하고 있다 [9]. 여기서 지적할 수 있는 상호운용성 시험의 한계는 시험기가 네트워크의 복잡한 논리적인 특성과 보다 중요한 시간적 특성을 내포하고 있지 못하고 있고 부분적인 시험으로 제한되어 있다는 점이다. 그 결과 위와 같은 상호운용성 시험은 적합성 시험을 IUT 별로 반복 수행한 것과 큰 차이가 없다는 것이다.

이를 극복하기 위하여 본 논문에서는 다음의 문제들의 해결 방안을 제시하고자 한다.

(1) 다수 공급자에 의해 제공된 통신망 구성 요소(즉, IUT)들의 기능적 및 성능적 특성과 IUT가 동작하는 주위 환경(위 예의 경우, 네트워크)을 실제 상황과 매우 가깝게 표현할 수 있는 방법이 있어야 한다.

(2) 위의 방법에 의해 표현된 통신망 구성 요소들의 상호운용성을 효과적으로 시험할 수 있는 플랫폼이 필요하다.

첫번째 문제를 해결하기 위한 접근 방법으로 실시간 객체 모형을 사용한다. 객체 지향 방법은 통신망 구성 요소들을 모듈화하고 자연스럽게 추상화하는 데에 매우 적합하다. 실제 통신망 구성 요소들을 정확하게 표현하기 위해서는 논리적인 상태의 변화뿐 아니라 시간적 특성이 고려된 모형이 필요한데, 기존의 객체 모형들로는 시간적 특성을 정확하게 효율적으로 표현할 수 없었다. 본 논문에서는 RTO.k(Real-Time Object by Kim and Kopetz) [5] 라고 하는 실시간 객체 모형을 이용하여 상호운용성 시험을 하고자 하는 대상(즉, IUT)들과 시험을 위한 주위 환경을 각각 시험 대상 객체(OUT: Object Under Test)들과 실제 시험 환경 시뮬레이션 객체(TESO: Test Environment Simulation Object)들로 나타낸다.

둘째로, 추상화(abstraction)된 실시간 객체들을 수행시킬 수 있는 실시간 보장 커널인 DREAM(Distributed Real-time Ever Available Microcomputing) 커널 [7]을 확장하여 시험을 위한 모니터 기법을 적용하여 플랫폼을 구축한다. DREAM 커널은 이더넷(ethernet)으로 연결된 PC상에서 운용된다. 각 노드에는 노드에 할당된 객체의 수행 상태를 감시, 제어하는 지역 모니터와 독립된 노드에 배치되어 시험 프로시저를 관리하고 수집된 자료를 분석하는 전역 모니터가 상호운용성 시험 임무를 분담한다.

실시간 객체 모형인 RTO.k와 수행 엔진인 DREAM 커널에 대하여 2장에서 설명한다. 3장에서는 상호운용성 시험을 위한 시험 구조와 이를 구현하기 위한 감시 메커니즘(mechanism)과 지역 모니터 및 시스템 전역 모니터의 기능을 제시한다. 여기서 제시된 시험 방법을 B-ISDN UNI 프로토콜에 시험 적용한 예를 4장에서 설명하고 5장에서 결론을 맺는다.

## 2. 실시간 객체 모형 및 수행 환경

이 장에서는 경성 실시간 시스템 개발을 위해 고안된 실시간 객체 모형인 RTO.k와 이 연산 객체들이 수행될 수 있도록 개발된 DREAM 커널에 대하여 소개한다.

### 2.1 실시간 객체 모형

실시간 객체 모형인 RTO.k는 다음과 같은 목표를 추구한다 [8].

- 경성 실시간 응용에서 비실시간 응용까지 적용할 수 있는 유연한 형태의 객체 모형
  - 실시간 커널 서비스와 실시간 객체 모형을 빌딩블럭화하여 실시간 서비스 설계 시 실시간성 보장
- 일반적 객체 지향 프로그램의 객체가 객체 데이터 공간(ODS: Object Data Store)과 멤버 함수들로 구성되는데 반해 RTO.k는 객체의 특성을 유지하면서 그 구성 요소가 데이터 공간과 동적인 멤버 쓰레드(메소드)로 구성되는 점이 다르다.

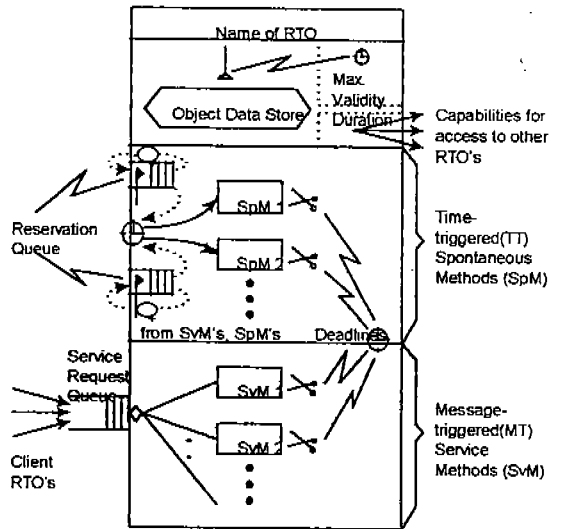
RTO.k의 중요한 두 가지 특징은 다음과 같다.

- (1) 두 가지로 분리된 메소드 그룹으로 구성된다. 첫 번째 그룹은 실시간 클럭에 의해 주기적으로 수행되는 시간 구동 메소드(time-triggered method)들로 SpM (Spontaneous Method)이라고도 한다. SpM은 시스템 설계 시에 그 작동이 결정되는 메소드이다. 작동 기간과 주기에 의해 수행되며 각 주기적 작동 시각에 대한 오차 허용 시간과 1회 작동 완료까지의 보장성 제한 시각이 주어진다. SpM의 이러한 시간 명세를 AAC(Autonomous Activation Condition)라 하며 다음과 같은 표현 방식을 갖는다.

"for t = from 10am to 10:50am every 2sec  
start-during t ± 10msec finish-by t + 0.5sec"

두 번째 그룹은 클라이언트로부터의 (설계 당시에는 알 수 없고 운용 중에 발생하는) 메시지에 의해 구동되는 메시지 구동 메소드(message triggered method)로 SvM(Service Method)으로도 불린다.

(2) SvM과 SpM의 구동이 공유되는 데이터 부분에 대해 충돌이 일어날 때에는 SpM에 우선권을 준다. 즉, SpM의 수행이 SvM에 영향을 받지 않도록 함으로써 설계 시 시간 보장을 용이하게 한 것이다. 이러한 규칙을 BCC(Basic Concurrency Constraint)라 하며 이는 객체 데이터 공간을 세그먼트로 나누고 각 세그먼트에 대한 메소드들의 접근 권한(access right)을 선언함으로써 가능하게 된다.



(그림 1) RTO.k의 구성 [5]

(Fig. 1) Structure of RTO. k (adapted from [5])

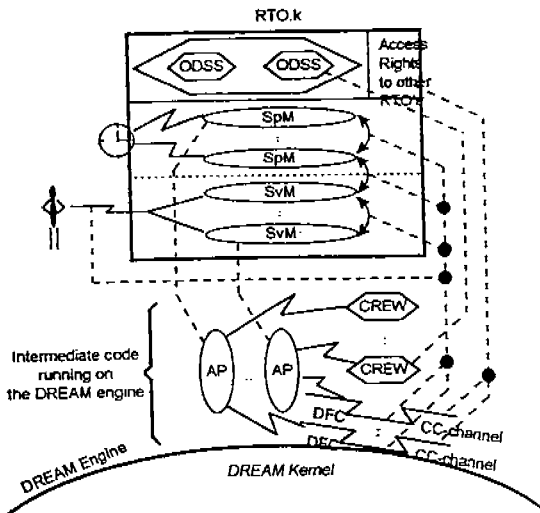
(그림 1)은 RTO.k의 구성을 나타낸다. 이 모형을 이용하면 실시간 시스템을 RTO.k 객체들의 연결망으로 표현할 수 있다. 각 객체들은 호출에 의해서 상호 작용한다. 호출은 클라이언트 객체의 SpM이나 SvM이 서버 객체의 SvM에 메시지를 보냄으로써 이루어진다. 클라이언트와 서버간에는 고도의 병렬 구동이 이루어지도록 비동기(non-blocking) 호출이 가능하며 서버 객체는 SvM이 정해진 제한 시간 내에 결과 출력을 보장한다.

2.2 DREAM 커널

RTO.k의 설계 시 시간 보장의 개념을 구현하려면 RTO.k 모형의 수행과 시간 보장 커널 서비스를 제공하는 실시간 커널이 필요하다. 미국 캘리포니아 주립 대학교(UC Irvine)에서 RTO.k의 수행 엔진인 DREAM 커널이 개발되었다 [6, 7]. 이 커널의 특징은 다음과 같다.

- 커널 서비스 시간의 예측을 위해 시간 점유의 우선 순위에 의한 5단계 계층으로 커널을 구성하여 RTO.k 객체의 SpM, SvM에 대한 실시간 스케줄링을 제공한다.
- 분산 IPC와 클럭 동기화에 의한 분산 환경용 커널로 커널 쓰레드에 의한 병렬성을 추구하였다.
- 커널 자신도 RTO.k 객체로 설계되었다.

RTO.k 객체 모형의 구성 요소들은 (그림 2)에 나타난 바와 같이 커널 상에서 프로세스 형태의 프로그램을 구성하는 요소들 간의 맵핑(mapping) 관계가 이루어진다. 즉, 객체 메소드는 프로세스로, ODS 세그먼트는 CREW(Concurrent Read Exclusive Write) 공유 데이터로, 그리고, 객체 간의 상호 작용의 위한 경로



(그림 2) RTO.k 객체 구성 요소와 프로세스 형태 프로그램 요소와의 상관 관계 [7]

(Fig. 2) Mapping of RTO.k object components onto process-type program components (adapted from [7])

는 CC(Content-Code) 채널로 맵핑된다. CC 채널은 논리적 멀티캐스팅 채널을 제공한다. SpM은 DREAM 커널에게 자신의 AAC와 제한 시간 그리고 ODS 접근 권한 등을 제시한다. SvM은 제한 시간, ODS 접근 권한 등을 제시한다.

DREAM 커널은 주어진 시간 제약 내의 서비스 양을 보장하는 실시간 커널로서, 인터넷으로 연결된 PC상에서 운용되며 DOS BIOS와 패킷 드라이버를 이용하고 프로그래밍의 노력을 덜어주는 라이브러리를 제공한다.

3. 상호운용성 시험 방법

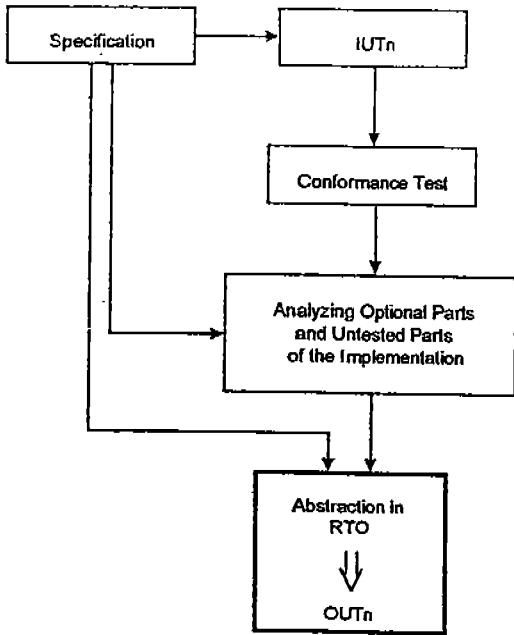
이 장에서는 RTO.k 객체 모형을 이용하여 상호운용성 시험을 하고자 하는 IUT와 주위 환경을 각각 시험 대상 객체(OUT: Object Under Test)들과 실제 시험 환경 시뮬레이션 객체(TESO: Test Environment Simulation Object)들로 나타내고, 이들이 수행되는 각 노드에는 노드에 할당된 객체의 수행 상태를 감시, 제어하는 지역 모니터와 독립된 노드에 배치되어 시험 프로시저를 관리하고 수집된 자료를 분석하는 전역 모니터로 상호운용성 시험을 하는 구조에 대하여 논한다.

3.1 시험 준비 절차

3.1.1 IUT의 실시간 객체화

시험하고자 하는 IUT는 객체가 지닌 캡슐화(encapsulation) 특성에 따라 자연스럽게 객체로 추상화될 수 있다. 2장에서 논의된 RTO.k 객체 모형은 통신 프로토콜 IUT의 논리적 및 시간적 특성을 표현하는데 매우 적합하다. IUT가 RTO.k 객체로 표현된 결과를 OUT라 한다. 어떤 IUT가 B-ISDN의 단말기라면 그 OUT의 ODS에는 그 구현의 상태를 나타내는 데이터가 있어야 하고, SpM은 그 상태를 갱신(update)하기 위한 동작을, SvM은 외부로부터 메시지를 받아 처리하는 내용을 나타내어야 한다. 통신 프로토콜에서 흔히 볼 수 있는 각종 타이머들은 RTO.k 객체의 실시간 클럭을 이용하여 표현할 수 있다. (그림 3)은 IUT에서 OUT를 위한 원시 자료를 도출하는 절차를 나타낸 것이다. 주어진 규격을 준수하는 n번째 구현물, 즉 IUTn이 제시되었고 이에 대한 적합성 시험이 실시되

있다면 IUTn에 대한 RTO.k 객체 표현, 즉 OUTn은 규격과 적합성 시험 결과를 토대로 생성된다. 적합성 시험 결과 나타난 IUTn의 논리적 시간적 특성을 반영한다. 규격의 프로파일은 크게 세 부분, (1)명백하게 정의된 필수 부분, (2)선택 사항으로 남은 부분, 그리고 (3)정의되지 않은 부분으로 나누어진다. 적합성 시험에 통과하였다면 필수 부분은 모두 만족 범위 내에 들게 된다. 적합성 시험 결과 나타난 선택 사항에 해당하는 데이터와 규격의 필수 부분 중 시험되지 않은 부분을 그대로 반영한다.



IUTn : nth Implementation Under Test  
 OUTn : nth Object Under Test  
 RTO : Real Time Object

(그림 3) OUT 원시 자료 도출 절차  
 (Fig. 3) OUT raw data obtaining procedure

이상의 과정을 거쳐서 OUT가 얻어지면 문제는 그것이 얼마나 정밀하게 본래의 IUT를 나타낼 수 있는가이다. 이에 대한 검증 방안으로 다음의 두 가지를 생각할 수 있다.

(1)적합성 시험 시뮬레이션: IUT에 대한 적합성 시험 내용을 시뮬레이션하여 OUT에 적용하고 얻어진 결과를 IUT 적합성 시험 결과와 비교한다.

(2)객체가 수행되는 PC와 적합성 시험기와의 인터페이스: IUT와 적합성 시험기 간의 물리적 인터페이스를 OUT가 구동되는 PC에 접합하여 시험하고 얻어진 결과를 IUT 적합성 시험 결과와 비교한다.

위의 두 가지 방법 중 PC 인터페이스가 간단하고 새로운 결함을 발생시킬 가능성이 적다면 두 번째 방법이 그렇지 않다면 첫번째 방법이 보다 효과적일 것이다.

### 3.1.2 시험 환경 시뮬레이션

상호운용성 시험을 가하고자 하는 IUT에 대한 RTO.k 객체 표현, 즉 OUT를 효과적으로 시험하기 위해서는 상호운용성 시험을 위한 환경도 RTO.k 객체 또는 객체들의 집합으로 표현되어야 한다. 이를 TESO 또는 TESO 그룹이라 한다. 시험 환경이 실제로 여러 물리 객체들로 구성되어 있거나, 하나의 객체로 하면 객체의 메소드 연산 처리량이 지나치게 많아서 연산 객체가 구동되는 PC의 연산 속도가 걸림돌이 되는 경우에는 시험 환경을 여러 객체들로 구성되는 TESO 그룹으로 모형화하고 객체들을 이더넷으로 연결된 여러 PC에 분산시킨다. 같은 모형에 의해 표현된 OUT와 TESO로 상호운용성 시험을 하면 실제 상황에서 순간적으로 발생하는 사건을 DREAM 커널이 제공하는 실시간 클럭의 한 간격(tick) 내에 발생한 것으로 처리할 수 있다.

B-ISDN의 여러 사용자 단말기 간의 상호운용성 시험을 한다고 하면 시험 환경은 하나 또는 연결된 여러 네트워크들과 시험 기준(reference)이 되는 사용자 단말기들이 될 것이다. 네트워크는 교환기에서 처리되는 복잡한 논리적 시간적 특성을 포함한 것이다. RTO.k 객체 모형은 이러한 복잡한 시험 환경을 매우 잘 추상화할 수 있다. 앞에 언급한 바와 같이 이 모형으로 시간적 특성을 잘 나타낼 수 있을 뿐 아니라 시험자가 원하는 정도까지의 모듈화를 통해서 시험의 정밀도를 조절할 수 있다. B-ISDN과 같은 실시간 분산 시스템의 특성을 객체 모형으로 나타내기 위해서는 이에 적합한 방법으로 시스템 분해가 이루어져야 한다. 이를 위해서 MDD(Multi-Dimensional Decomposition) 방법을 적용한다 [11]. 이는 본래 저자 등이 포함된 연구 그룹에서 객체 지향 시스템 설계 방법의 일환으로 고안된 것이다. 이 방법을 적용하면

시험 환경을 구성하는 객체들을 도출하기 위해서 이를 (1)동작 모드별, (2)수평적으로 유사한 작업을 수행하는 객체 그룹, (3)수직적으로 사건 발생 경로를 따라 관련된 객체 그룹, (4)도출된 객체 그룹을 다시 데이터, 동작 형태 등에 따라서 세분하는 과정을 거치게 된다. MDD 방법에 의해 도출된 객체들은 다시 시험의 목적에 따라 보다 적은 수의 객체들로 합쳐지게 된다. 시험 목적에 관련되지 않은 부분에 대해서는 시간적 특성만을 고려한다. B-ISDN 네트워크를 TESO로 표현하면 TESO의 ODS에는 처리 중인 각 호의 상태와 네트워크가 유지해야 할 정보가 저장, 유지된다.

3.2 상호운용성 시험 구조

상호운용성 시험을 가하는 OUT들과 TESO 그룹, 그리고 시험을 위한 관찰, 제어, 시험 프로시저 관리를 행하는 전역 모니터 객체의 연결망을 (그림 4)에 나타내었다. 모든 객체는 RTO.k 모델을 따르고 동일한 플랫폼 상에서 수행된다. 플랫폼 하드웨어는 이더넷으로 연결된 PC이고 플랫폼 소프트웨어는 DREAM

커널과 프로그래밍 지원 서비스, 그리고 지역 모니터로 구성된다.

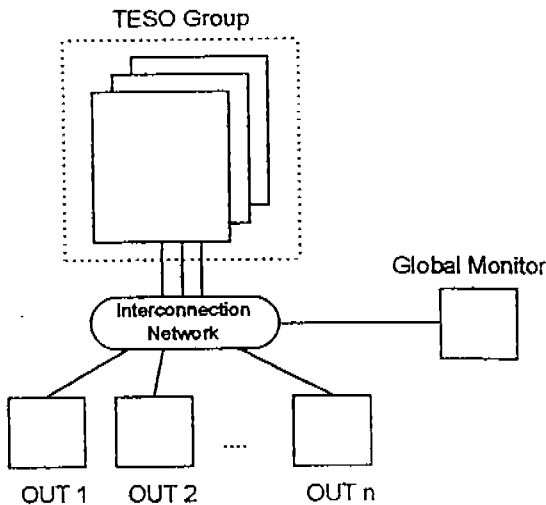
객체의 수행 상태를 효과적으로 감시하기 위한 방안으로 객체(메소드)간 메시지 흐름(message flow)의 모형[3]을 이용한다. 객체 내 메소드는 메시지 통신에 의해서 상호 작용하는데, 한 메소드에서 발생한 메시지와 이것에 연결된 메소드의 연속된 고리를 메소드 체인이라고 한다. 즉, 일련의 연속된 수행에 관련된 메소드와 메시지의 집합이다. 하나의 메소드 체인은 한 메소드에서 시작하여 한 개 또는 그 이상의 다른 메소드로 연결될 수 있으며 더 이상 메시지를 발생시키지 않는 메소드에 이르게 되면 멈춘다. 메시지는 목적지 객체와 객체 내 해당 메소드의 이름과 메소드 입출력 변수 등을 포함한다.

플랫폼은 새로 생성된 객체에게 고유 번호를 부여하고, 객체 수행을 스케줄링하며, 원래 요청된 메시지에 제어 정보를 부가하여 얻어진 확장 메시지를 목적지 객체에 전송한다. 확장 메시지 내에는 그 메시지의 감시 상태를 제어하는 정보가 들어 있다. 해당 지역 모니터는 이 정보를 검사하여 어떤 외부 메소드 체인이 추적되어야 하는가를 확인할 수 있다. 메시지 전달이나 객체 스케줄링과 같은 외부 수행을 감시한다. 감시 중인 메소드 체인의 경우, 각 메소드 수행 처리 시간과 입출력 값의 범위를 확인하고 이들 데이터를 저장한다. 위급한 상태가 아닌 경우, 전역 모니터의 지시에 따라 저장된 데이터를 모아서 전역 모니터에 전달한다. 확장 메시지 내의 제어 정보를 이용하여 선택적으로 외부 객체 수행을 감시할 수 있다.

각 지역 모니터의 주요 임무는 다음과 같다.

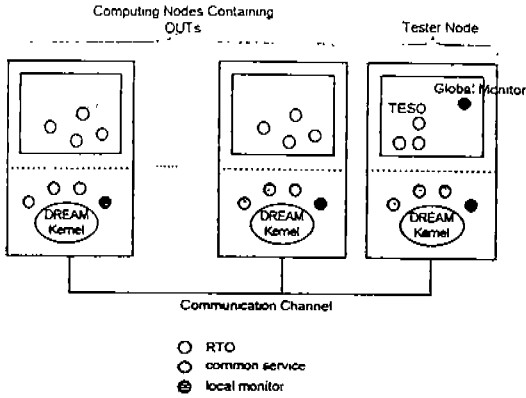
- 해당 노드에 배치된 객체들의 수행 상태를 감지하여 각 객체들의 상태에 대한 정보를 유지한다.
- 전역 모니터로부터 감시해야 할 대상과 조건을 부여 받고 해당 객체를 제어, 관찰한다.
- 얻어진 정보를 저장하였다가 전역 모니터에 보고한다.
- 객체 수행 중 발생하는 예외 상황을 전역 모니터에 보고한다.

전체 시스템 내에는 OUT를 포함하는 컴퓨팅 노드들과 시험 노드가 있다. (그림 5)에는 RESO 그룹과 전역 모니터가 함께 시험 노드에 배치된 예를 나타내고 있다.



TESO : Test Environment Simulating Object  
 OUT : Object Under Test

(그림 4) 상호운용성 시험을 위한 실시간 객체 연결망  
 (Fig. 4) Network of objects for interoperability test



(그림 5) 논리적 객체 연결망의 컴퓨팅 노드 배치 예  
(Fig. 5) Example of mapping logical object network on computing nodes

전역 모니터의 기능은 다음과 같다.

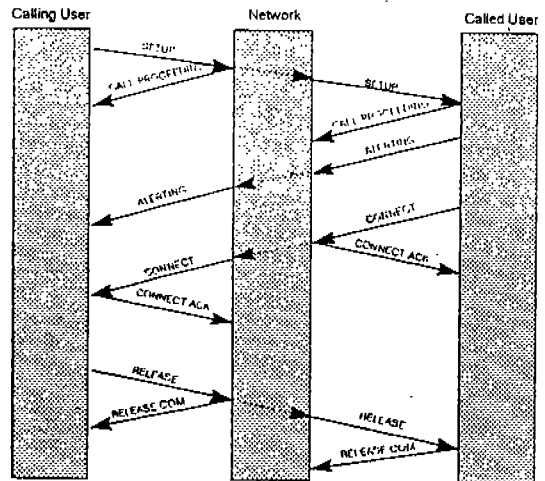
- 응용 서비스 제공자로부터 제시된 응용 프로그램의 엄밀한 규격을 바탕으로 유효한 입력 조건과 유효한 사건 발생 순서 등을 확인한다.
- 감시해야 할 객체와 메소드 체인의 유형과 감시해야 할 시간 등의 감시 조건을 명시한 시험 프로시저를 생성하고 관리한다.
- 각 컴퓨팅 노드의 지역 모니터와의 메시지 통신을 통하여 해당 시험 프로시저를 제공하고 감시 서버들에 의해 수집된 감시 데이터를 저장 분석한다.
- 자동 시험을 위하여 시험 사건을 발생시키고 동시에 시험 프로시저를 진행시킨다.

감시 데이터를 분석하여 시스템 스펙트를 확인하여 고유 번호를 붙여서 관리하고 분석 결과를 시각화할 수 있다. 시험 프로시저는 시험 사건의 종류와 이를 보내는 순서를 정의한다. 순차 프로그램을 위한 일반 시험 소프트웨어는 시험자에 의해 제어되는 프로시저로 충분하다. 그러나, 자치적으로 수행되는 다수의 분산 객체의 경우 시험자는 여러 시험 사건들을 보내서 이러한 객체들을 제어해야 한다. 어떤 시험 사건들은 순차적으로 또 다른 것들은 동시에 비동기적으로 보내져야 한다. 시험자의 운용 오류를 방지하기 위하여 시험은 자동으로 수행될 필요가 있다.

#### 4. 적용 사례

이 장에서는 3장에서 제안한 상호운용성 시험 구성 방안을 B-ISDN UNI Q.2931 프로토콜 상호운용성 시험에 적용한 결과에 대하여 논한다.

B-ISDN 망 요소로는 단말장치와 네트워크를 생각할 수 있다. 간단한 예로 두 개의 단말 장치가 교환기를 포함하고 있는 네트워크를 통해서 통신하고 있는 경우 양단 간의 호 연결 제어 메시지 흐름을 (그림 6)에 나타내었다. 종전의 UNI 상호운용성 시험은 두 개의 단말장치에 대한 시험이 대부분이었다. 그러나, 3장에서 제시된 방법을 적용하면 OUT들로 추상화된 여러 단말장치들이 순차적으로 또는 비동기적으로 동시에 작용하는 경우에 대한 상호운용성 시험을 수행할 수 있다.



(그림 6) B-ISDN 호 연결 제어 메시지 흐름도  
(Fig. 6) Control message flow for B-ISDN call connection

(그림 7)은 여러 사용자 단말장치의 상호운용성 시험을 위해 RTO.k 객체로 추상화된 OUT들(즉, USER들)과 시험 환경을 시뮬레이션하는 TESO(즉, NETWORK)를 나타낸다. OUT의 ODS에는 각 호의 상태와 단말장치에 필요한 정보가 저장된다. 실시간 시뮬레이션 클럭 간격 내에 SpM에 의해 단말장치의 상태가 갱신된다. SvM에 의해 네트워크로부터 메시지를 받아 두거나, 지역 모니터로부터 시험을 위한 제어 메시지를 받는다. TESO의 ODS에는 진행 중인 모든 호의 상태와 네트워크에서 관찰하는 기타 정보가 저

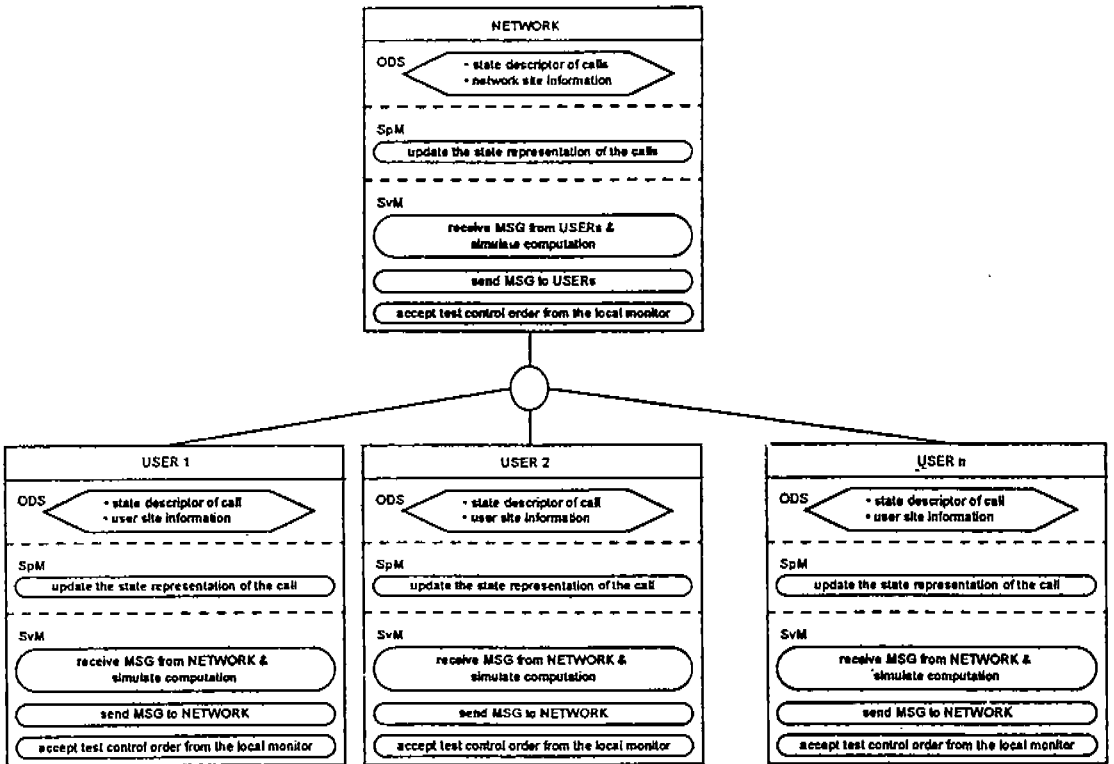
장된다. OUT의 경우와 마찬가지로 SpM에 의해 상태가 갱신되고, SvM에는 여러 사용자 단말장치, 즉, OUT들과 해당 지역 모니터로부터 시험 제어 서비스 요청이 입력된다. 시스템 전역 모니터와 TESO를 한 노드에 그리고 세 개의 OUT를 각각 노드에, 모두 네 개의 노드에 배치하고 간단한 실험을 행하였다. 그 결과 이 방법이 다음과 같은 강점을 지니고 있음을 확인하였다.

- 실시간 시뮬레이션 클럭을 사용, 클럭의 한 간격 내에 TESO 그룹과 OUT들 간의 상호작용이 완료 될 수 있도록 함으로써 IUT와 시험기를 분리할 때 발생하던 동기화 문제를 해결하였다.
- 여러 IUT에 대한 순차적인, 비동기적인 시험을, 동적으로 수행할 수 있으며 시험 과정을 자동화할 수 있다.

- 여러 OUT들 중에서 일부 OUT의 행동에 대한 선택적인 시험이 가능하다. 따라서, 집중적인 반복 시험을 통하여 높은 결함 발견율을 얻을 수 있다.
- TESO 그룹의 객체 들과 OUT 그룹의 객체들이 그 경계를 약간의 수정 만으로 쉽게 넘나들 수 있기 때문에, 어떤 OUT의 상호운용성 시험이 완료되었다면 이 객체를 TESO 그룹에 포함시켜서 다른 시험을 계속할 수 있어서 시험의 확장이 매우 용이하다.

### 5. 결 론

실시간 객체 모형에 기반을 둔 상호운용성 시험 접근 방법을 제시하였다. 실시간 객체 모형은 통신 시



(그림 7) B-ISDN UNI 프로토콜 상호운용성 시험 적용 예  
(Fig. 7) Example of applying interoperability test to B-ISDN UNI protocol



시스템을 구성하는 시험 대상 구현물이나 시험 환경을 매우 잘 나타낼 수 있다. 이러한 객체들을 실시간 커널 상에서 실행시키고 상호운용성 시험을 위한 관찰과 제어 기능을 시스템 전역 모니터와 각 노드에 내장(embedded)된 지역 모니터에 분담하여 통신망과 시험 노드의 부담을 완화하는 방법을 적용하였다.

이 방법은 시험 확장성이 우수하고 여러 IUT 간의 동적인 시험을 가능케 한다. 설계 시에 검증을 위한 수단으로도 활용될 수 있고, 물리적으로 구현된 제품에 대해서도 시험에 요구되는 정밀도에 따라 시뮬레이션할 수 있다. 보다 실질적인 상호운용성 시험 도구로 쓰이기 위해서는 규격이나 구현물로부터 실시간 객체를 효과적으로 도출할 수 있는 지원 도구의 개발이 요구된다.

참 고 문 헌

[1] 최진영, 홍범기, "프로토콜 시험 기술 및 B-ISDN 시험을 위한 고려 사항", 한국전자통신연구소 주간기술동향, 94. 6.

[2] ISO 9646-1, "Information Technology-Open system Interconnection-Conformance Testing Methodology and Frame Work", part 1-5, 1994.

[3] P. C. Jorgensen and C. Erickson, "Objected-Oriented Integration Testing", Comm. of the ACM, Vol.37, No.9, pp.30-38, Sept. 1994.

[4] J.Joyce, G. Lomow, K.Slind, and B.Unger, "Monitoring Distributed Systems", ACM Trans. Computer Systems, Vol.5, No.2, pp. 121-150, May 1987.

[5] Kane Kim and H.Kopetz, "A Real-Time Object Model RTO. k and an Experimental Investigation of Its Potentials", Proc. COMPSAC '94, Taipei, Nov. 1994.

[6] Kane. Kim, "Realization of Autonomous Decentralized Computing with RTO.k Object Structuring Scheme and HU-DF Inter-Process-Group Communication Scheme", Proc. IEEE Computer Society's 2nd Int'l Symp. on Autonomous Decentralized Systems, pp. 305-312, Apr. 1995.

[7] Kane Kim, L. Bacellar, Y. Kim, C. Subbaraman,

H. Yoon, Jungguk Kim and K. W. Rim, "A Timeliness-Guaranteed Kernel Model-DREAM Kernel and Implementation Techniques", RTCSA '95, Tokyo, Oct. 1995.

[8] 김문희 외, "실시간 시스템 모형", 한국정보과학회지, 14권 8호, pp.15-21, 96. 8.

[9] O. Rafiq and R. Castanet, "From Conformance Testing to Interoperability Testing", IFIF Protocol Test Systems, III, pp. 371-385, 1991.

[10] G.S. Vermeer and H. Blik, "Interoperability Testing: Basis for the Acceptance of Communication Systems", IFIF Protocol Test Systems, VI, pp. 315-330, 1994.

[11] S.M. Yang, et. al., "System Development based on a Real-Time Object Model", IEEE WORDS96, pp.152-159, 1996.



최진영

1991년 한양대학교 산업공학과 졸업  
 1993년 2월 한국과학기술원 산업공학과 공학석사 (산업공학)  
 1993년 2월~현재 한국전자통신연구원 연구원

관심분야: 프로토콜 적합성 시험, 상호운용성 시험, B-ISDN



민병준

1983년 연세대학교 전자공학과 졸업(학사)  
 1985년 연세대학교 대학원 전자공학과 졸업(석사)  
 1991년 미국 캘리포니아주립대 (UC Irvine) 전기컴퓨터공학과(박사)

1984년~1986년 삼성전자 종합연구소  
 1992년~1993년 한국통신 연구개발원  
 1994년 감사원 전산담당관실  
 1995년~현재 시립인천대학교 전자계산학과 재직중  
 관심분야: 분산실시간 시스템, 결합허용, 통신망관리



김 문 회

1979년 서울대학교 전기공학과  
학사

1981년 2월 서울대학교 전기공  
학과 석사

1985년 5월 University of South  
Florida, MSCS

1991년 5월 University of Cali-  
fornia, Berkeley, Ph.D

1991년 3월~현재 건국대학교 컴퓨터공학과 부교수  
관심분야: 소프트웨어공학, 운영체제, 실시간 분산처  
리시스템, 통신망관리