

# DEVS 형식론 기반 멀티쓰레드 소프트웨어 개발 방법을 이용한 스토리지 서버 프로그램 구현

임정현\*, 오하령\*\*, 성영락\*\*

\*국민대학교 보안-스마트 전기자동차 학과

\*\*국민대학교 전자공학부

e-mail: ijh227@kookmin.ac.kr, hroh@kookmin.ac.kr, yeong@kookmin.ac.kr

## Embodiment of Storage Server Program using Multi-threaded Software Development Process based on DEVS formalism

Jung-Hyun Im\*, Ha-Ryoung Oh\*\*, Yeong-Rak Seong\*\*

\*Dept of Secured-Smart Electric Vehicle, Kookmin University

\*\*College of Electrical Engineering, Kookmin University

### 요 약

최근 스토리지 서버에 사용되는 코어가 싱글코어에서 점차 발전하여 멀티코어가 됨에 따라 스토리지 서버에 많은 기능들이 추가되었다. 이러한 기능들을 효과적으로 사용하기 위해서는 스토리지 서버를 효율적으로 관리하는 프로그램이 필요하다. 이에 멀티코어를 효과적으로 사용하여 스토리지 서버를 효율적으로 관리할 수 있도록 멀티쓰레드 프로그램으로 스토리지 서버 프로그램을 구현하였다. 멀티쓰레드 소프트웨어는 동시 동작으로 인해 개발하는데 어려움이 있으므로 이를 해결코자 DEVS 형식론 기반의 멀티쓰레드 소프트웨어 개발 방법을 이용하였다. DEVS 형식론 기반의 모델링과 시뮬레이션을 거치고 소프트웨어를 구현하여 개발의 어려움과 검증에 대한 부분을 해결하였다.

### 1. 서론

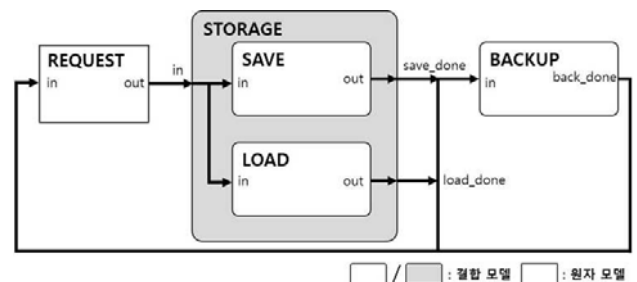
최근 클라우드 서비스의 발전에 따라 스토리지 서버의 연구가 활발히 진행 중이다. 스토리지 서버의 CPU의 코어수가 싱글코어에서 멀티코어로 증가하였고 이를 효율적으로 사용할 수 있는 프로그램이 필요하게 되었다. 또한 기술의 발전에 의해 많은 기능들이 추가 되었다 [1]. 이에 멀티코어를 이용하여 스토리지 서버를 효율적으로 사용할 수 있는 멀티쓰레드 소프트웨어 스토리지 서버 프로그램을 구현하였다. 멀티쓰레드 소프트웨어는 동시에 여러 쓰레드를 구동하여 여러 가지 작업을 병행 수행할 수 있으므로 사용자에 대한 응답성을 높일 수 있는 장점이 있다. 그러나 이러한 장점에도 불구하고 쓰레드의 동시 동작으로 인해 소프트웨어 개발에서 테스트와 디버깅에 대한 어려움이 있었다. 이러한 어려움을 해결하고자 DEVS 형식론[2] 기반의 멀티쓰레드 소프트웨어 개발 방법을 이용하여 스토리지 서버 프로그램을 멀티쓰레드로 구현하였다.

### 2. 스토리지 서버 프로그램 구현

스토리지 서버 프로그램을 멀티쓰레드로 구현하기 위해 DEVS 형식론 기반의 멀티쓰레드 소프트웨어 개발 방법[3-4]을 이용하였다. 멀티쓰레드의 동시 동작으로 인한 멀티쓰레드 소프트웨어 개발에서 생겼던 테스트와 디버깅의 어려움을 DEVS 형식론 기반의 모델링과 시뮬레이션을 통하여 해결하였다. 또한 DEVS 형식론의 추상화 시뮬

레이터인 DEVSim++ 시뮬레이터를 사용해 구현한 스토리지 서버 프로그램을 시뮬레이션을 수행하여 검증하였다.

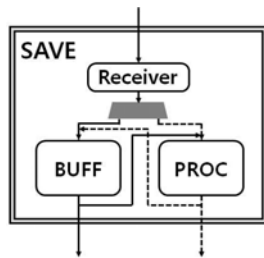
스토리지 서버 프로그램 구현은 DEVS 형식론 기반의 멀티쓰레드 소프트웨어 개발 방법에 따라 네 단계로 구현하였다. 첫 번째 단계로 스토리지 서버 프로그램의 요구사항을 분석하였다. 본 논문에서는 스토리지 서버 프로그램의 모든 기능을 구현하기 보다는 멀티쓰레드로 기본적인 스토리지 서버 프로그램의 기능들을 구현하여 효율적인 프로그램의 개발이 가능함을 보였다. 그러므로 본 논문에서 정의한 스토리지 서버 프로그램은 기본적인 스토리지 서버 프로그램의 동작으로서 사용자의 데이터 저장, 불러오기 명령을 수행한다. 또한 스토리지 내에 저장된 데이터의 백업을 병행한다.



(그림 1) 모델링 된 스토리지 서버 프로그램 모델

두 번째 단계는 DEVS 형식론 기반의 모델링과 시뮬레이션을 수행한다. 모델링은 DEVS 형식론을 기반으로 하여 (그림 1)과 같이 최상위 결합모델인 STORAGE\_SERVER 모델 하위에 사용자의 명령을 전달하는 REQUEST 원자모델이 있다. 또한 사용자의 저장 명령을 처리 하는 SAVE 결합모델과 불러오기 명령을 처리 하는 LOAD 결합모델을 포함하는 STORAGE 결합모델, 저장된 데이터를 백업하는 BACKUP 결합모델로 이루어진다. 각각의 SAVE, LOAD, BACKUP 결합모델 내부에는 명령에 해당하는 데이터를 저장하는 BUFF 원자모델과 전달받은 명령에 해당하는 동작을 수행하는 PROC 원자모델로 이루어진다.

모델링 결과는 DEVS 형식론 기반의 추상화 시뮬레이터(abstract simulator)인 DEVSim++ [5]를 이용하여 시뮬레이션 하였다. 시뮬레이션 결과 모델링 된 모델은 첫 번째 단계에서 정의한 기능들에 대한 동작을 충실히 수행하였다.

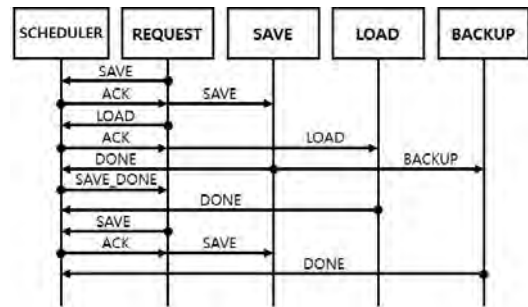


(그림 2) 구현한 SAVE 쓰레드 내부 구조

세 번째 단계에서는 모델링 된 모델을 편평화하고 편평화 된 모델을 쓰레드로 구현하였다. 쓰레드는 총 5개 원자 모델로부터 생성된 원자 쓰레드들의 동작을 관리하는 SCHEDULER 쓰레드가 있으며, REQUEST 원자모델을 구현한 쓰레드와 SAVE, LOAD, BACKUP 결합모델이 각각 하나의 쓰레드로 구현된다. 각각의 SAVE, LOAD, BACKUP 결합 모델의 BUFF 원자모델과 PROC 원자모델을 합쳐 하나의 쓰레드로 구현하므로 구현된 쓰레드는 (그림 2)와 같이 두 모델의 입력을 받는 하나의 수신부와 각 모델의 동작 내용을 가진 처리부로 구성된다. 그리고 [4]에서 제안된 쓰레드 수 조절방법에 따라 연관성이 있는 BUFF, PROC 모델을 하나의 쓰레드로 구현할 때 쓰레드 내에서 바로 해당 모델로 전달 할 수 있도록 연관된 두 모델간의 데이터 전달을 함수 호출로 구현하였다. 함수 호출을 통한 데이터 전달 시 데이터는 함수의 인자로서 전달된다.

네 번째 단계에서는 구현된 소프트웨어의 동작 순서와 DEVSim++를 이용한 시뮬레이션 결과의 모델 동작 순서를 비교하여 구현된 스토리지 서버 프로그램이 부합하게 동작하는지 확인하였다. 비교 결과, 추상화 시뮬레이터를 이용한 시뮬레이션은 가상 시간(virtual time)을 기반으로 하고 있어 소프트웨어의 동작과 완벽히 같지는 않았지만 스토리지 서버 프로그램의 동작 순서는 (그림 3)과 같이

시뮬레이션의 모델의 동작 순서와 부합하게 동작하였다.



(그림 3) 스토리지 서버 프로그램 동작 순서

### 3. 결론

본 논문에서는 멀티쓰레드로 구동되는 스토리지 서버 프로그램을 구현하였다. 멀티쓰레드의 동시 동작으로 인한 개발의 어려움을 DEVS 형식론 기반의 멀티쓰레드 소프트웨어 개발 방법을 이용하여 해결하였다. 또한 DEVS 형식론 기반의 시뮬레이션과 소프트웨어의 동작을 비교함으로써 구현된 소프트웨어를 검증하였다. 검증 결과 소프트웨어의 동작이 시뮬레이션과 적합하게 동작하는 것을 확인하였다.

구현한 소프트웨어를 바탕으로 기능별로 쓰레드를 생성하여 동시에 여러 작업을 수행할 경우 스토리지 서버의 성능이 향상 될 것이다. 또한 클라우드 서비스의 상용화에 따라 스토리지 서버 프로그램의 중요성이 부각되므로 네트워크 관련 기능에 대한 연구가 필요하다.

### 참고문헌

- [1] WANG, Yuedui, et al, "The content delivery network system based on cloud storage," In: Network Computing and Information Security (NCIS), 2011 International Conference on. IEEE, 2011. pp. 98-102, 2011.
- [2] Bernard P. Zeigler, Object-Oriented Simulation with Hierarchical, Modular Models, Academic Press, 1990.
- [3] Y.H. Kim, Y. R. Seong, and H. R. Oh, "Software Development Method Using the Concurrency Control Approach Based on DEVS Simulation," The 2014 FTRA International Conference on ACS, pp. 7-10, 2014.
- [4] 임정현, 오하령, 성영락, "DEVS 형식론에 기반한 멀티 쓰레드 소프트웨어개발 과정에서의 쓰레드 수 조절," 한국정보통신학회 2015년도 동계종합학술발표회, pp. 1373-1374, 2015.
- [5] T.-G. Kim, "DEVSim++ User's Manual : C++ Based Simulation with Hierarchical Modular DEVS Model," Computer Engineering Lab., Dept. of Electrical Engineering, KAIST, 1994.