

# 시뮬레이션 도구 SMPLE의 개발 및 활용

조성만, 조경산

단국대학교 전산통계학과

## 요 약

컴퓨터 시스템의 개발 및 분석에 많이 활용되는 시뮬레이션 모델을 구축하는데 있어서 SIMSCRIT, GPSS, SIMAN, SLAM II 등과 같은 시뮬레이션 전용언어가 많이 사용되지만, 이들은 새로운 전용언어의 습득, 시스템 프로그래밍과의 접촉, 시뮬레이션 수행 속도면에서 부담을 주어 왔다.

본 연구에서는 기존의 C언어 사용자들이 손쉽게 시뮬레이션 모델을 구축할 수 있도록 기존의 `smpl`을 확장하여 SMPLE(`smpl extended`)를 개발하였다. SMPLE의 모체인 `smpl`은 컴퓨터 시스템의 시뮬레이션 모델을 구현하기 위해 MIT에서 C언어를 이용하여 고안되었으며, C언어의 기능적인 확장으로서 라이브러리 함수들의 집합 형태를 갖는 시뮬레이션 서브 시스템이다. 이러한 라이브러리 함수들의 집합인 `smpl` 시뮬레이션 서브 시스템은 C언어 자체와 더불어 사건중심(event-oriented) 시뮬레이션 언어를 구성하며, `smpl` 시뮬레이터는 C언어 프로그램으로 구현된다.

`smpl`은 시뮬레이션하기 위한 설비(facility)들을 정의, 예약, 해제하거나 상태를 알아보는 함수를 제공하며, 시간의 관점에서 보면 토큰의 흐름은 일련의 대기시간과 활동시간에 의하여 나타낼 수 있게 된다. `smpl`은 사건의 스케줄링(scheduling), 확률변수의 생성, 통계자료 수집 등에 관한 함수의 제공과, 시뮬레이션 시간의 전진과 사건발생의 순서를 조절해주는 기능을 제공한다. `smpl` 시뮬레이션 프로그램은 초기화루틴, 제어루틴 및 결과출력루틴으로 구성된다. 기존의 `smpl`에서 사용되는 프로그램의 자료구조(data structure)는 배열과 인덱스 구조가 사용되었다. 이러한 구조는 이미 완성되어있는 프로그램 내에서는 효율성 면에서 좋겠지만, 변경이나 확장하기 위해 이 프로그램을 분석한다거나 필요한 기능을 추가하기 위해서는 매우 많은 어려움이 따르게 된다.

본 논문에서는 배열을 기반으로 하고 있는 `smpl`의 자료구조를 C의 structure와 pointer를 기반으로 하게끔 변경시키고 이에 따르는 제반 변경 사항을 수정 보완하여 프로그램의 분석을 용이하게 하며 기능의 변경 및 추가가 수월하게 하였고 메모리를 동적으로 관리할 수 있게 하였다. 또한 기존의 `smpl`에 디버깅용 함수 및 설비(facility) 제어용 함수를 추가하여 시뮬레이션 프로그램 작성을 용이하게 하였다. 예를 들면 `who_server()`, `who_queue()`, `pop_Q()`, `push_Q()`, `pop_server()`, `push_server()`, `we()`, `wf()`, `printfct()` 같은 함수들이다. 또한 동시에 발생하는 사건들의 순서를 조종하기 위해, 동시에 발생할 수 있는 각각의 사건에 우선순위를 두어 이 우선순위에 의하여 사건 리스트(event list)에서 자동적으로 사건들의 순서가 결정되도록 확장하였으며, 설비 제어방식에 있어서도 FIFO, LIFO, 우선 순위 방식등을 선택할 수 있도록 확장하였다. SMPLE는 자료구조 및 프로그램이 공개되어 있으므로 프로그래머가 원하는 기능을 쉽게 추가할 수 있는 장점도 있다.

아울러 SMPLE에서 새로이 추가된 자료구조와 함수 및 설비제어 방식등을 활용하여 실제 중형급 시스템에 대한 시뮬레이션 구현과 시스템 분석의 예를 보인다.