HLA Application 개발을 위한 CASE Tool

박민호 김재형 정창성
고려대학교 전자공학과
{minopak, true}@snoopy.korea.ac.kr csjeong@charlie.korea.ac.kr

CASE Tool For HLA Application

Min-Ho Park, Jae-Hyung Kim, Chang-Sung Jeong
Dept. of Electronics Engineering, Korea University

요 약

HLA(High Level Architecture)는 차세대 DIS(Distributed Interactive Simulation)를 이어 네트워크 분산 시뮬레이션의 기반이 되는 표준 아키텍처이다. 이 HLA 시스템의 개발 과정에서 FEDEP(Federation Development and Execution Process) Model이 제안되었는데, FEDEP의 목적은 federation 개발자가 application의 요구를 충족시킬 수 있도록 federation 개발 및 실행에 대한 guideline을 정하는 것이다. FEDEP의 일련의 process들 중에서 노동집약적인 과정은 CASE(Computer Aided Software Engineering) tool을 사용함으로써 보다 강화되고 능률적으로 이루어질 수 있다. 본 논문에서 소개하는 HLA Application Builder 는 federate 와 federation을 자동적으로 생성하는 CASE tool 로서, 자동적인 FED(Federation Execution Data)file 및 C++ source code를 생성함으로써 HLA Federation 개발에 있어서의 인력(manpower)을 크게 줄일 수 있다. 본 논문에서는 HLA에 대한 배경지식과 HLA Application Builder 개발의 필요성과 구현, 그리고 실제 예를 들어서 HLA Application Builder가 어떻게 federation 개발에 사용되는지에 대해서 설명한다.

1. Background

HLA는 미국의 DoD(Department of Defense)의 산하기관인 DMSO(the Defense Modeling and Simulation Office) 를 중심으로 1993년 IEEE (Institute of Electrical and Electronic Engineers) 표준으로 제정됨과 더불어 나아가 ISO 표준으로 진전되기를 희망하고 있으며 이미 국방분야의 시뮬레이션에서 HLA는 필수적인 항목으로 고려되고 있다.

HLA는 네트워크 가상환경 구축을 위한 고 수준 시뮬레이션 아키텍처로서 네트워크에 연결된 서로 다른 형태인 federate들 간의 상호작용을 통하여 하나의 가상세계인 federation을 재현하다.

이는 차세대 DIS 표준의 기반으로 네트워크를 이용한 군사용 모의 시뮬레이션 시스템의 구성요소, 디자인 룰, 인터페이스 등에 관한 전반적인 아키텍처로도 정의할 수 있다. 아키텍처의 목적은 시뮬레이션간의 상호연동성과 시뮬레이션 컴포넌트의 재사용성(reusability)을 높이기 위한 것이다. HLA는 Object Model Templates(OMT), Interface Specification 인 Run-Time Infrastructure (RTI), HLA Compliance Rules 의 3가지 구성요소를 가지고 있다.[1]

·Object Model Templates(OMT): federation에 존재할 개체와, 개체의 속성, federation내에서 개체간의 상호작용 등의 정보를 선언해둔다.

·Run-Time Infrastructure(RTI): federate들 간의 상호작용을 지원하는 Interface Specification으로 실제 데이터 교환과 시스템 조정을 담당한다.

·HLA Compliance Rules: HLA를 따르는 시뮬레이션이 되기 위해 필수적으로 따라야 하는 규칙이다.

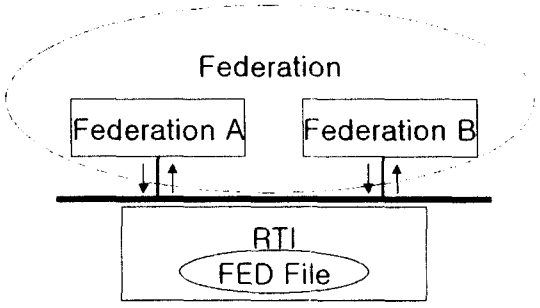

그림 1 HLA Overview

Object Model Templates에서 정의된 object들이 모여서 하나의 federate를 이루고, 이 federate는 HLA

Compliance Rule에 따라서 RTI를 통해 다른 federate 들과 상호작용을 하게 된다. 이런 federate들이 모여서 하나의 federation이 된다. RTI는 federation에서 사용되어지는 객체와 event 등을 미리 정의해 놓은 FED (Federation Execution Data)file을 가지고 federate 들 간의 통신을 담당한다.

2.FEDEP Model

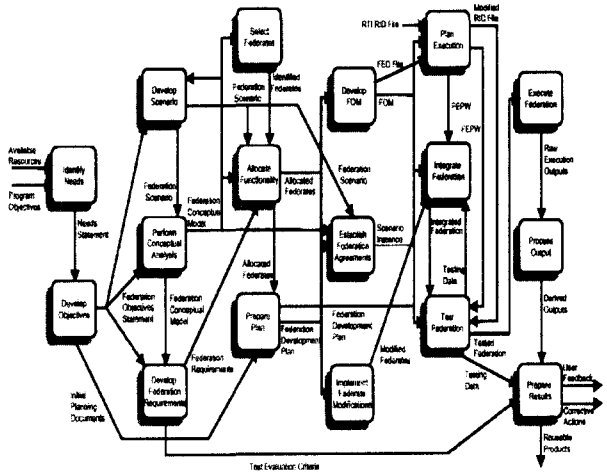

그림 2 FEDEP Model

FEDEP 모델은 HLA federation의 개발과 실행을 위하여 High-level framework를 기술한다. 그림2는 FEDEP 모델의 전체 process를 도식화한 것이다. 이 모델의 목적은 HLA application 개발자가 개발작업을 보다 효율적으로 할 수 있게 guideline을 제공하는 것이다.[1]

FEDEP에서 제안된 일련의 과정 중에 상당한 수작업을 필요로 하는 process가 있다. 이 노동집약적인 process를 효율적으로 처리하면서 application을 개발하기 위해서 자동화 프로그램(automated software tool)이 강조되고 있다. DMSO에서 배포하고 있는 OMDT(Object Model Development Tool)가 대표적인 예이다.

OMDT는 OMT(Object Model Template)를 작성하는 형태로 만들어져 있는데, OMT 는 federation의 효율적인 개발을 위해 필요한 정보를 일정한 형식의 table로 만들어 놓은 것이다. OMDT는 federation을 개발하는 과정에 있어서 OMT와 같은 table 형식을 사용하여 FEDEP 모델중 일부 과정을 처리하는 tool이다. 작성된 table을 통해서 FED 파일이 생성된다. 이밖에 상용소프트웨어로도 자동화 tool의 개발이 계속되고 있다.

3. HLA Application Builder

3.1. 개발의 필요성

FEDEP Model에 따라서 federation을 개발하는 것은 일반적으로 상당한 노동력이 들어가는 작업이다. 개발작업을 인력(manpower)으로 수행하는 중에 발생하는 복잡함과 어딘가에 숨어있을지 모르는 오류를 줄여야 한다. 또 작성된 OMT를 이용해서 효율적으로 federation을 개발 할 수 있어야 하고, 기존에 만들어진 federation을 재사용 할 수 있게 함으로써 HLA application 개발에 도움이 되어야 한다.

3.2. Design & Implementation

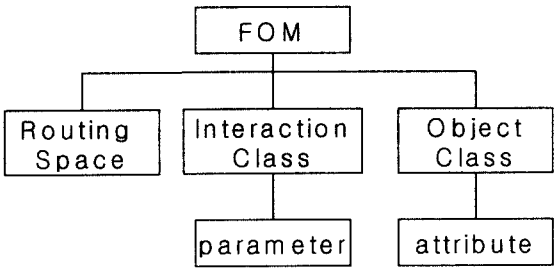

그림 3 Federation Object Model

FOM(Federation Object Model)은 federation에서 정의되는 Object Model로서 그림과 같이 구성되어 있다.

Object Class에서는 object를 정의해준다. 정의된 object 가 federation에서 사용될 수 있고, 각각의 object는 필요한 attribute를 가진다. Interaction Class에서는 federation에서 발생할 수 있는 event를 정의해준다. 또 각 event마다 필요한 parameter를 선언해준다. Routing Space는 federate사이의 object attribute와 interaction parameter의 흐름에 대한 선언이다.

HLA application Builder는 Object class, attribute, Interaction class, parameter를 정의함으로써, 하나의 application이 만들어지기 위한 기본적인 framework을 만들어 준다. windows98/2000에서 동작하고, Visual C++로 구현했으며, 기본적인 윈도우즈 GUI를 사용했다. 개발자는 hierarchical한 구조의 Object Class 와 Interaction Class를 tree 형식의 표에 추가, 삭제, 수정을 함으로써 FOM을 만들게 된다. 여기서 만들어진 FOM을 이용해서 federation에서 필요한 FED파일과 C++ source code를 자동적으로 생성하게 된다.

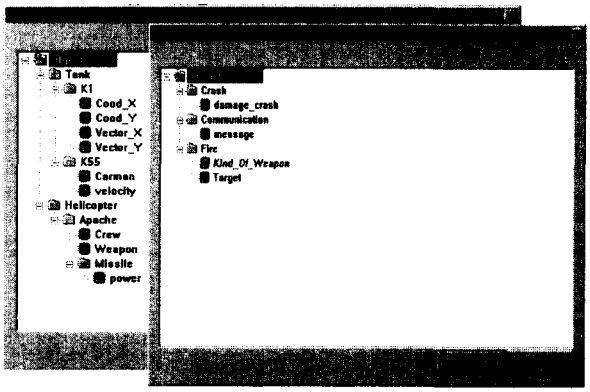

그림4 Object Class 와 Interaction Class의 예

3.3. Example

헬기 대 전차 교전 모델을 예로 들어보자. 먼저 전장(battlefield)에서는 헬기와 전차가 존재한다. Object Model Class에서 이것을 정의해준다. Tank의 종류는 K1과 K55가 있고 각 Tank의 attribute를 정의한다. Helicopter class도 마찬가지로 정의한다. Interaction Class는 전장에서 Object들 사이에 발생하는 event를 정의해 주는 class이다. 여기서는 Crash(두 객체간 충돌), Communication(두 객체간 통신), Fire(무기발사) 의 3 개를 예로 들었다. Crash Interaction은 두 객체가 충돌할 때 발생하며, 이때 damage_crash로 정의된 parameter가 각 federate들에게 파괴정도를 전달한다. Communication Interaction은 federate 들 사이의 메시지를 전달할 때 발생하며, 전달할 내용을 message 라는 parameter로 전달한다. Fire Interaction은 무기를 발사할 때 일어나는 interaction으로, 무기의 종류와 목표물을 parameter로 전달하게 된다. 이렇게 FOM이 구성되고 나면 FED file이 생성되게 된다. RTI는 이렇게 생성된 FED 파일을 이용해서 federation에 참여한 federate 간의 통신을 담당하게 된다. HLA Application Builder는 이러한 일련의 분석 과정을 통해서 만들어진 FOM을 이용해서, federation이 만들어지기 위한 framework를 C++코드로 자동 생성해 주게 된다.

4.결론 및 향후 발전 방향

FEDEP Model 은 federation을 개발하는데 있어서 guideline을 정해 놓은 것이다. 본 논문에서는 FEDEP Model을 따라서 federation을 개발할 때 요구되는 상당한 인력을 줄이고 빈번하게 나타나는 오류와 복잡함을 없애기 위해서 HLA application Builder 라는 CASE

```
;; User Object Classes Here
        (class Tank
                (class K1
                        (attribute Cood_X)
                        (attribute Cood_Y)
                .....
                )
        (class Helicopter
                (class Apache
                        (attribute Crew)
                        (attribute Weapon)
                        (class Missile
                                (attribute power)
                        )
                )
        )
;; User Interaction Classes Here
        (class Crash
                (parameter damage_crash)
        )
        (class Communication
                (parameter message)
        )
        (class Fire
                (parameter Kind_Of_Weapon)
                (parameter Target)
        )
```

그림5 생성된 FED파일

(Computer Aided Software Enginnering) tool의 개발 과정과 구현에 대해서 설명하였다. HLA application Builder는 Object class 와 Interaction class를 선언해서, FED file과 federation의 framework 가 되는 C++ source code를 생성함으로써 효율적인 federation 개발이 이루어질 수가 있다.

본 논문에서 소개한 HLA application Builder는 federation 개발의 60% 정도를 자동적으로 수행해 준다. 앞으로 routing space 처리부분을 첨가하고, 입력 부분의 세밀한 구분을 통해서 개발 과정 전체를 자동으로 처리할 수 있도록 발전 시켜야 한다.

5.참고문헌

[1]Defense Modeling and Simulation Office, "High Level Architecture Federation Development and Execution Process(FEDEP) Model version 1.5," December 8, 1999.

[2]Defense Modeling and Simulation Office, "High Level Architecture Interface Specification," April 2 , 1998.