

모의기반획득을 위한 시물레이션 아키텍처 프레임워크 개발

조규태^{1†} · 심준용¹ · 이용현¹ · 이승영¹ · 김세환¹

Development of Simulation Architecture Framework for Simulation Based Acquisition

Kyutae Cho · Junyong Shim · Yongheon Lee · Seungyoung Lee · Sachwan Kim

ABSTRACT

Recent modeling and simulation technology is being used in various fields. Especially in the field of military, Simulation-Based Acquisition (SBA) is recognized as a essential policy. To effectively carry out SBA, modeling and simulation techniques should be applied in the whole life-cycle for the weapon system development, and simulation architecture framework which provides easily reusable and interoperability is needed. Through reusability and interoperability, the costs of constructing the integrated collaborate environment for simulation based acquisition can be minimized. In this study, we define requirements, issues for enhancing reusability and interoperability, and propose simulation framework as a solution of the problem including structural design. Proposing simulation framework provides common functions for producing simulator as reusable units and easily changeable structure on user's purpose. In addition, we provide the result for applying simulation framework to our project.

Key words : Framework, SBA, Reusability, Interoperability, Architecture

요 약

최근 모델링 및 시물레이션 기술이 다양한 분야에서 활용되고 있으며 특히, 국방 분야에서는 모의기반획득(Simulation Based Acquisition)이 필수적인 정책으로 인식되고 있다. 모의기반획득을 효율적으로 수행하기 위해서는 무기체계 개발의 전 순기에 모델링 및 시물레이션 기법을 적용할 수 있어야 하고 이를 위해서 모의기 하부 컴포넌트 재사용 및 상호운용을 위한 시물레이션 아키텍처 프레임워크가 필요하다. 재사용 및 상호운용을 통해 모의기반획득 환경 구축을 위한 비용을 최소화 할 수 있다. 본 연구에서는 재사용성 및 상호운용성 강화를 위한 프레임워크 에 대한 요구사항 및 문제점을 정의하고 그 해결책으로 개발한 시물레이션 프레임워크의 설계구조를 제시한다. 제안하는 시물레이션 프레임워크는 시물레이터를 구성하는데 공통적으로 필요한 기능을 재사용의 단위로 제공하며 사용자가 필요에 따라 쉽게 변경하거나 추가할 수 있다. 이러한 프레임워크를 통해 모의기반획득 환경 구축을 위한 비용을 최소화 하는 것이 가능하다. 제안하는 프레임워크를 실제 사업에 적용한 결과를 제시하고 다양한 적용방안 및 향후 연구 방향에 대하여 설명한다.

주요어 : 프레임워크, 모의기반획득, 재사용성, 상호운용성, 아키텍처

1. 서 론

최근 모델링 및 시물레이션(M&S) 기술이 다양한 분야에서 활용되고 있으며, 특히 국방 분야에서는 모의기반획

득(SBA; Simulation Based Acquisition) 이 필수적인 정책으로 인식되고 있다^[1]. 모의기반획득은 무기체계 소요 분석 및 결정에서 분석/설계, 제작, 시험/평가, 훈련/운용/군수지원에 이르는 전 과정 상에서 M&S를 활용함으로써 무기체계 획득의 시간, 자원, 위험을 실질적으로 감소시킬 수 있다. 무기체계 개발의 전 순기에 M&S 기법을 적용한다는 것은 각 획득 단계 별 M&S의 적용을 통한 협업을 의미한다^[2]. 모의기반획득의 중요한 요소인 통합협업 환경이란 M&S의 재사용성, 상호운용성, 표준화 등을 증진

2010년 7월 2일 접수, 2010년 9월 7일 채택

¹⁾ LIG넥스원 Maritime연구센터

주 저 자 : 조규태

교신저자 : 조규태

E-mail; kyutcho@lignex1.com

하는 시스템이다^[3]. 재사용성을 높이기 위해서는 각 단계의 M&S 컴포넌트를 공유하고 실행하기 위한 공통 기반 환경의 구축이 필요하다. 아키텍처 프레임워크를 통하여 M&S 컴포넌트의 공통 기반 환경구축이 가능하고 각 컴포넌트의 단계별 재사용이 가능하게 된다.

공통 아키텍처 프레임워크가 필요한 또 다른 배경으로 는 방대해진 시뮬레이션 환경을 들 수 있다. 무기체계 및 정보체계가 점점 복잡해지면서 이를 반영하기 위해 무기체계의 M&S 환경 또한 네트워크 기반의 분산시뮬레이션 환경이 일반화되며 복잡해졌다. 분산 시뮬레이션 환경으로 인하여 하드웨어적으로는 높은 처리성능을 확보하였지만 그만큼 모의 소프트웨어의 규모가 증가하였고 이는 각 모의 소프트웨어의 중복 개발 및 개발오류 가능성 또한 증가시키는 결과를 초래한다. M&S 프레임워크를 통해서 이러한 모의 소프트웨어 중복 개발 및 오류를 줄일 수 있다.

또한 네트워크상에서 모든 전투세력들을 실시간으로 연결하여 전투력을 향상시키는 이른바 네트워크 중심전(NCW; Network Centric Warfare)개념이 일반화되면서 무기체계간의 상호운용성을 위한 요구도 증가하였다. 이에 따라 무기체계를 검증하기 위한 시뮬레이터간 상호운용성이 중요해졌고 이를 제공하기 위한 시뮬레이션 아키텍처 개발도 활발히 진행되고 있다.

본 연구에서는 M&S 컴포넌트의 재사용성(Reusability)을 향상시키고 상호운용성(Interoperability)을 보장하는 시뮬레이터 제작을 위한 기반 환경인 시뮬레이션 아키텍처 프레임워크를 제안한다. 제안하는 프레임워크는 무기체계 개발 전 순기에 사용될 수 있는 기반기능을 제공함으로써 효과도 분석을 위한 일체형 시뮬레이션부터 실시간 분산 시뮬레이션을 통한 실무기체계 연동 시뮬레이션(HILS; Hardware In-the-Loop Simulation) 구성까지 폭넓게 적용될 수 있다. 이때 재사용성과 상호운용성을 보장함으로써 모의기반획득 통합협업 환경구축을 위한 비용을 최소화 할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 프레임워크 및 상호운용성 관련 연구를 소개하고, 3장에서는 제안하는 시뮬레이션 프레임워크 설계내용을 기술한다. 4장에서는 프레임워크 적용 사례와 그 결과를 제시한다. 5장에서는 결론 및 향후 연구방향을 기술한다.

2. 관련 연구

앞에서 살펴본 것처럼 프레임워크를 통한 공통 기반환

경은 M&S 컴포넌트의 재사용을 통해 각 개발 단계별 협업을 증진시키고 중복개발 및 개발오류를 감소시키는 장점을 가지고 있다. 또한 NCW 환경에서의 무기체계 간 연동을 위한 상호운용성의 기반을 제공할 수 있다. 이러한 M&S 아키텍처 프레임워크를 개발하기 위해 국내 및 국외에서 많은 노력을 기울여왔다.

미군은 Joint Program Office에서 1998년에 Joint Modeling and Simulation System(JMASS)를 개발하였고 지속적인 성능 개선 및 적용을 추진하고 있다^[4]. JMASS는 개발 완료된 하나의 시뮬레이션 시스템이 아니라, 시뮬레이션을 지원하는 통합 환경 및 도구의 집합이다. 따라서 각종 모델, 모델 인터페이스 및 시뮬레이션 엔진 등을 포함하는 아키텍처 프레임워크이다. JMASS의 기본 모델 단위는 Player이며, 시뮬레이션 단위는 Team으로 이루어진다. 이때 Team은 단지 조직화만 수행하므로 실제 JMASS에서 시뮬레이션 행위의 기본은 Player이다. 각 Player 간의 통신은 Port를 통해 이루어지며 Port를 통해 연결될 때만 다른 시뮬레이션 등에서 사용할 수 있는 상호운용성을 보장할 수 있다. Player, Port, Team은 기본적으로 C++ 객체이므로, 재사용은 ‘코드’ 수준에서 이루어진다^[5]. 따라서 JMASS에서 컴포넌트의 재사용은 plug-and-play방식이 아니라 plug-and-rebuild 방식으로 이루어진다. 또한 JMASS의 모델링 개념에 일치하지 않는 HLA서비스에 대해서는 접근이 제한되어 명시적인 시간 증가가 어려운 점이 존재한다.

PlugIn-Based Architecture(PBA)는 컴포넌트간 상호 독립적인 인터페이스 및 상호 작용이 가능함으로써 기존 아키텍처에 비해 컴포넌트의 추가 및 조합이 용이하고 재사용성이 높은 구조를 위해 제안되었다^[6]. PBA를 사용하면 시나리오에 따라 컴포넌트를 plug-and-play방식으로 재구성할 수 있으며 시뮬레이션 실행 시에도 구성품의 생성/삭제가 가능하도록 고안하였다. 이를 위해 사건 관리, 스케줄링, 동기화를 담당하는 시뮬레이션 엔진과 시뮬레이션 S/W를 구성하는 빌딩블록인 PlugIn이 존재한다. 각 PlugIn은 PlugIn Adapter를 통하여 데이터를 송수신 할 수 있고 이때 데이터의 송수신 규약은 PlugIn Object Model(POM)을 통하여 미리 규정되어 야 한다. PBA는 JMASS에 비해 컴포넌트의 유연한 재사용을 지원할 수 있지만 HILS 구성을 위한 실시간 제약을 만족하기 위해서는 PlugIn간의 통신 최적화가 필요하다.

미 국방성(DoD)은 분산 시뮬레이션 환경의 상호운용성을 보장하고 모의 시뮬레이터를 포함한 실시스템과의

연동을 용이하게 하며, 시뮬레이션 소프트웨어의 재사용성을 촉진시키기 위한 방안으로 High Level Architecture (HLA)를 발표한다^[7]. HLA는 분산 연합 시뮬레이션을 위한 공통기반 아키텍처를 제공할 뿐 단일 시뮬레이터 자체에 대한 아키텍처를 포함하지 않으며 시뮬레이션 엔진 등의 기반 서비스는 제공하지 않는다^[6]. 따라서 시뮬레이터 컴포넌트 단위의 재사용이 어렵다.

3. 시뮬레이션 프레임워크

시뮬레이션 프레임워크를 설명하기 전에 먼저 일반적인 개념을 정의해 보면 소프트웨어 프레임워크는 특정 응용분야 소프트웨어에 대해서 재사용할 수 있도록 설계된 상호작용하는 클래스(Class)들의 집합이라고 할 수 있다^[8]. 여기서 재사용은 재정의의 통하여 이루어 질 수 있으며 프레임워크 재정의란 프레임워크에 정의한 클래스를 상속받아 특정 응용프로그램을 지원하는 서브클래스를 정의하는 것이다. 이러한 재정의의 과정을 통해서 해당 영역에서 새로운 응용프로그램을 만들 수 있고, 이때 프레임워크가 정의한 추상 클래스를 상속하여 새로운 클래스를 정의하는 방법을 사용한다. 즉, 공통의 영역에 존재하는 응용프로그램들의 가변성을 제외하고 일반적인 설계 과정에 대한 결정을 해놓음으로써 재사용의 기반을 마련한다. 프레임워크와 응용프로그램과의 관계를 살펴보면 그림 1과 같다.

시뮬레이션 아키텍처 프레임워크는 시뮬레이션 소프트웨어 작성을 위한 기반을 제공함으로써 시뮬레이터 컴포넌트의 재사용성을 향상시키는데 그 목적이 있다. 따라서 시뮬레이션 아키텍처 프레임워크를 통하여 다음의 기능을 제공한다.

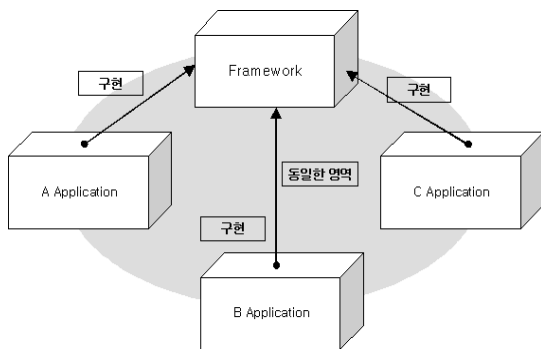


그림 1. 프레임워크와 어플리케이션의 관계

- 시뮬레이션 구성품의 기본적인 설계 제공
- 프레임워크를 확장하여 응용 프로그램 개발 가능
- 시뮬레이션 시스템의 기능단위 컴포넌트 제공
- 컴포넌트 재사용을 위한 컴포넌트 통신 표준 제공
- 시뮬레이션 환경의 상호운용성 제공
- 시뮬레이션 진행을 위한 시뮬레이션 엔진 제공

시뮬레이션 시스템의 기능단위 컴포넌트 제공을 위하여 시뮬레이터를 구성하는 컴포넌트를 기능 단위로 정의하였다. 그림 2에서 분산 시뮬레이션의 예를 들어 기능 단위를 나타낸다.

그림 2와 같이 운용통제 장비는 설정, 관측, 통제, 시나리오, 통신 관련 기능이 필요하고 표적 시뮬레이터와 유도탄 시뮬레이터는 설정, 모의, 통제, 파라미터, 시나리오, 통신 기능이 필요하다. 따라서 이러한 시뮬레이터 제작에 필요한 전체 기능 단위를 식별하고 식별된 기능을 재사용 컴포넌트인 ‘관리자’(manager)로 정의하였다. 표 1에서 현

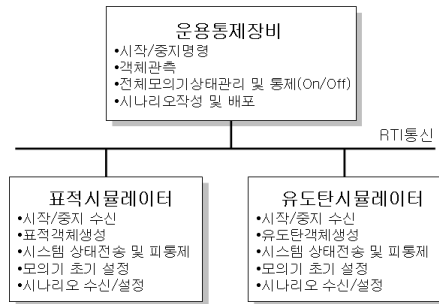


그림 2. 분산 시뮬레이터 기능 예시

표 1. 시뮬레이터 기능 및 관리자

기능	관리자	설명
설정	설정통제 관리자	시뮬레이터 상태 관리 및 관리자 간 메시지 송수신 제어
통제	통제관리자	원격 구성품 제어 및 상태정보 점진 기능 제공
시나리오	시나리오 관리자	시나리오 데이터 관리를 위한 읽기/수정/저장 수행
파라미터	파라미터 관리자	장비 초기설정정보 관리
모의	모의관리자	모의 정보 생성 및 관리
관측	UI관리자	GUI와 Backend 간의 인터페이스 기능 제공
통신	시뮬레이션망 관리자	HLA/RTI 미들웨어 인터페이스 제공

표 2. 시뮬레이션 엔진 구성 및 기능

구성모듈	기능
메시지 디스패처	관리자 간 메시지-이벤트 등록 및 분배
스케줄러	실시간 시뮬레이션 진행 기능 제공

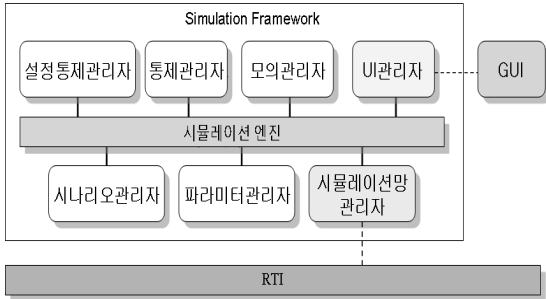


그림 3. 시뮬레이션 프레임워크 구성

재 정의된 시뮬레이션 기능단위 및 관리자를 설명하였다.

이러한 관리자들은 서로간의 결합도를 낮추고 재사용성을 향상시키기 위하여 시뮬레이션 엔진이 제공하는 메시지 기반 인터페이스로 통신한다. 또한 시뮬레이션 엔진을 통하여 이벤트를 정의하고 실시간으로 시뮬레이션을 진행할 수 있다. 시뮬레이션 엔진의 세부 모듈 및 기능은 표 2와 같다. 시뮬레이션 엔진에서 제공하는 기반 인터페이스를 사용하는 컴포넌트 구현을 통하여 사용자가 원하는 기능의 관리자를 추가하는 것이 용이하다.

시뮬레이션 프레임워크의 전반적인 구성은 그림 3과 같다. 관리자 간의 내부통신은 시뮬레이션 엔진을 통해 구현되고 UI관리자를 통해 관측을 위한 외부 GUI와 통신을 수행하며 시뮬레이션망 관리자를 통해 다른 시뮬레이터와의 RTI 통신을 수행한다.

시뮬레이션 엔진 및 각 관리자의 구성을 자세히 살펴 보도록 한다.

3.1 시뮬레이션 엔진

시뮬레이션 엔진은 시뮬레이션을 진행하여 시뮬레이터의 상태를 변화시키는 기능을 수행한다. 이러한 시뮬레이션 엔진은 시간의 흐름에 따른 진행을 담당하는 스케줄러(Scheduler)와 메시지 전달에 따른 진행을 담당하는 메시지 디스패처(Message Dispatcher)로 구성되어 있다. 시뮬레이션 엔진의 구성은 그림 4와 같다.

메시지 송신을 원하는 관리자는 제공된 인터페이스를 통해 메시지 디스패처에게 메시지를 전달하고 수신 관리자가 등록된 등록 이벤트를 콜백하여 이벤트를 수행한다.

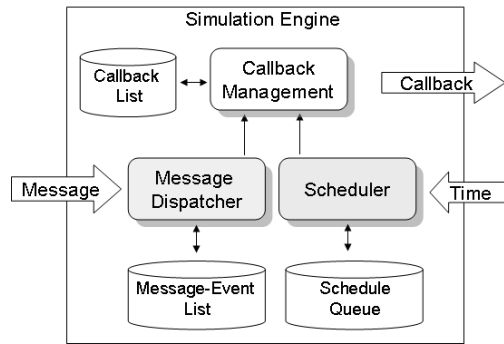


그림 4. 시뮬레이션 엔진 구조

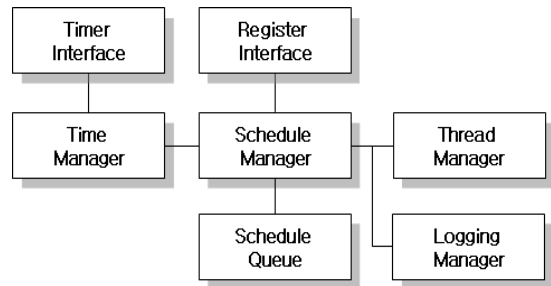


그림 5. 스케줄러 구조

메시지 디스패처를 사용함으로써 프레임워크의 사용자는 일관된 연동 인터페이스를 유지할 수 있고 이는 관리자의 재사용성을 높이고 시뮬레이터 구조를 쉽게 이해할 수 있게 한다.

시간에 의한 시뮬레이션 진행을 위해서 시뮬레이션 엔진은 스케줄러 모듈을 제공한다^[9]. 그림 5는 스케줄러 구조를 나타낸다.

스케줄러는 수행될 작업을 정의하는 Register Interface, 수행될 작업을 저장하는 Schedule Queue, 현재 시간의 작업을 수행하는 Schedule Manager, 시간을 최소단위로 진행시키는 Time Manager로 구성된다. 이때 시간의 측정 방법은 Timer Interface를 통해서 쉽게 변경할 수 있다. 현재 시뮬레이션 프레임워크에서는 실시간 시뮬레이션 진행을 위해 GPS Board Timer와 Multimedia Timer를 사용하고 있다. 또한 사용자는 Register Interface를 통해 ‘정기적인 반복 수행’ 또는 ‘일정 시간 후 수행’ 등으로 작업 패턴을 정의할 수 있다. 시간측정 방법의 최소 진행 단위 만큼 시간을 증가시키며 Schedule Queue에 현재 작업 패턴에 따라 수행되어야 하는 작업들이 있는 경우 멀티 쓰레드를 사용하여 작업을 진행한다. 스케줄링이 수행되는 동안 스케줄링 지연 발생 여부를 모니터링하여 사

용자가 참조할 수 있도록 하는 보고서 작성 기능도 제공한다. 스케줄러 실행 흐름도는 그림 6과 같다.

3.2 시뮬레이션망 관리자

시뮬레이션망 관리자는 시뮬레이션 프레임워크의 HLA/RTI 통신 기능을 담당하는 모듈로서^[10], 표준 연동 모델인 RPR-FOM(Real-Time Platform Reference-Federation Object Model)^[11]을 적용하여 시뮬레이션 연동을 지원한다. RPR-FOM은 SISO(Simulation Interoperability Standards Organization)에서 육·해·공군의 공통 플랫폼 객체를 정

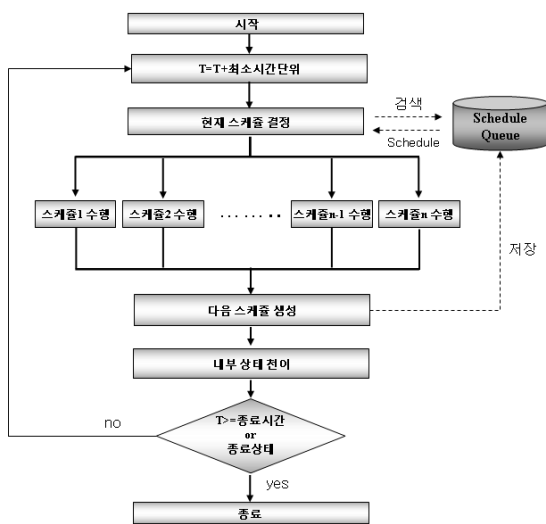


그림 6. 스케줄러 실행 흐름도

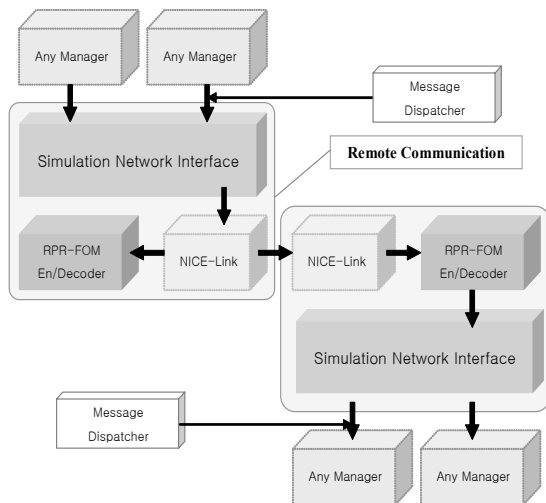


그림 7. 시뮬레이션망 관리자 구조

의한 표준 FOM이다. 시뮬레이션망 관리자는 크게 3개의 모듈로 구성되며, 그림 7과 같이 메시지 디스패처를 통해서 프레임워크의 내부 관리자들과 통신한다.

3.2.1 Simulation Network Interface

시뮬레이션 네트워크 인터페이스는 프레임워크의 내부 및 외부 메시지를 연결시킨다. HLA 기반의 다른 프레임워크를 개발할 경우 해당 모듈만 수정하면 시뮬레이션 네트워크 관리자의 재사용이 용이하다.

3.2.2 NICE-Link

NICE-Link^[12]는 RTI 서비스의 사용자 편의성을 향상시키기 위한 모듈로서, 시뮬레이션 통신 기능을 제공하는 프레임워크이다. 이 모듈은 RTI가 사용하는 메타 정보 및 복잡한 서비스를 캡슐화하고 있다. 또한 Federate의 객체 모델인 SOM(Simulation Object Model)을 해석하여 연동 정보에 대한 선언 서비스를 자동으로 수행한다.

3.2.3 RPR-FOM Encoder

분산 객체를 송·수신하기 위해서 객체를 바이트 단위로 변환하여 저장 및 복원할 수 있는 직렬화(Serialization) 작업이 필요하다. RPR-FOM Encoder는 시뮬레이션 프레임워크에서 사용하는 표준 객체인 RPR-FOM의 직렬화 기능을 구현한 모듈이다.

이처럼 시뮬레이션망 관리자는 RTI서비스와 RPR-FOM의 인코딩 및 디코딩 모듈을 제공함으로써 분산 시뮬레이션 환경 구축을 위한 기반 및 상호운용성을 제공한다.

3.3 시나리오 관리자

시나리오 관리자는 시뮬레이션 프레임워크 내의 시나리오에서 사용되는 데이터의 관리를 담당한다. 이는 XML(Extensible Markup Language) 기반의 공격/방어 시나리오와 각 무기체계의 제원으로 이루어진 데이터베이스를 포함한다. 시나리오 데이터 관리를 통해 분산 환경의 다른 장비와 시나리오 데이터를 교환할 수 있는 기능을 제공한다. 시나리오 관리자의 기능은 다음과 같다.

- XML 파일 형태의 시나리오를 해석(Parsing)하여 프레임워크에서 사용하는 형태로 변환
- 시나리오 데이터를 XML 파일 형태로 저장
- 시나리오 데이터를 RTI를 통해 전달할 수 있는 형태로 인코딩
- RTI를 통해 전달받은 시나리오 데이터를 원래의 형

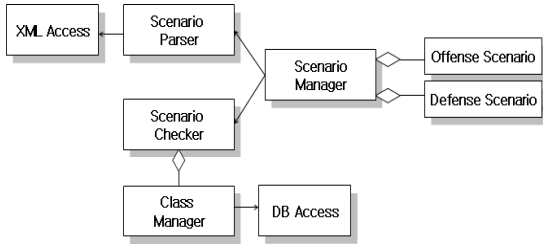


그림 8. 시나리오 관리자 모듈 관계도

테로 디코딩

- DB에 저장되는 각종 시나리오 데이터의 생성/수정/삭제를 관리
- 시나리오 데이터의 구성요소들 간의 무결성 및 값 검사

시나리오 관리자의 역할 중 핵심적인 기능은 시나리오의 Parsing이다. XML 포맷의 파일과 Database 형태로 이루어진 시나리오를 시나리오 관리자의 Parser 모듈에서 프레임워크에서 사용할 수 있는 형태의 데이터로 Parsing 해주는 역할을 한다. Parsing된 시나리오 데이터는 메시지 디스패처를 통하여 다른 관리자로 전달되며 이 데이터는 다시 해당 관리자의 역할에 따라 시뮬레이션에 사용되거나 시뮬레이션망 관리자를 통하여 다른 분산 시뮬레이션 환경의 시뮬레이터에 전달할 수 있게 된다. 시나리오 관리자는 시뮬레이션 프레임워크의 구성 요소로서 프레임워크가 적용된 모든 시뮬레이터 및 통제장비에 공통적으로 사용된다.

시나리오 관리자는 위에서 제시한 기능을 모듈 단위로 구현하고 있다. 각 모듈간의 의존성을 최소화함으로써 모듈의 변경으로 인한 다른 모듈의 수정이 적은 유연한 구조를 가지고 있다. 예를 들어 시나리오의 내용이 변경되는 경우가 발생하면 시나리오 Parsing 모듈과 저장 모듈을 변경에 맞게 수정하기만 하면 되고 다른 모듈은 일체 수정할 필요가 없다. 시나리오 관리자 모듈 관계는 그림 8과 같다.

3.4 모의 관리자

모의 관리자는 시뮬레이션 프레임워크 내에서 모의 객체 및 모의 정보 생성을 위한 기반 클래스를 제공한다. 모의 관리자는 시뮬레이션 엔진의 스케줄러를 사용하여 주기적으로 반복되는 모의 정보계산을 수행하고 모델의 상태를 변경할 수 있다. 생성되는 모의 정보는 시뮬레이션

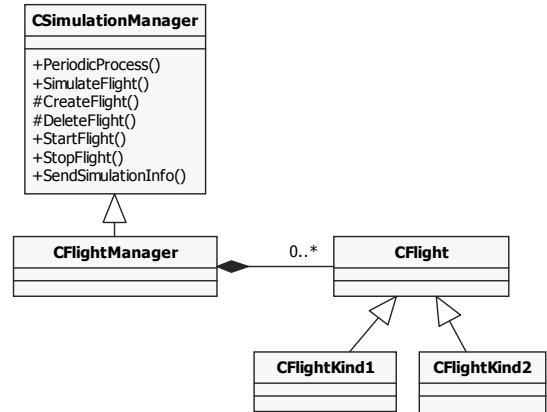


그림 9. 표적모의기 개략 클래스도

망 관리자에게 전달되어 RPR-FOM에 정의된 객체로 변경된다. 그림 9는 모의 관리자를 상속받아 다양한 종류의 표적을 만들기 위한 표적모의기의 개략 클래스도이다.

3.5 통제 관리자

통제 관리자는 분산 환경을 구성하는 장비의 원격 점검 및 제어를 위한 기능을 제공한다. 이를 위하여 다음의 두 가지 기능을 제공한다.

- CBIT(Continuous Built-In-Test)
- RBIT(Request Built-In-Test)

CBIT은 분산 환경에 참여하는 시뮬레이터 또는 실장비의 상태를 확인하는 기능이다. RBIT은 시뮬레이터에 문제가 발생한 경우 문제의 원인이 무엇인지 요청하는 기능이다. RBIT은 CBIT과 연계하여 장비에 문제가 생긴 경우 수행한다.

통제 관리자는 통제를 하고자 하는 장비와 피통제를 위한 장비에서 모두 사용되어야 한다. 피통제 장비의 통제관리자가 점검한 내용을 메시지 디스패처를 통하여 시뮬레이션망 관리자에 전달하면 시뮬레이션망을 통하여 통제장비로 전달되어 전시가 가능하게 된다.

3.6 설정통제 관리자

설정통제 관리자를 통하여 프레임워크 내부 관리자들에게 전달되는 각종 메시지를 저장하고 전체 모의기의 상태를 관리할 수 있다. 또한 관리자간에 연결되는 메시지 수신지점을 단일화하여 메시지 디스패처 연결을 단순화

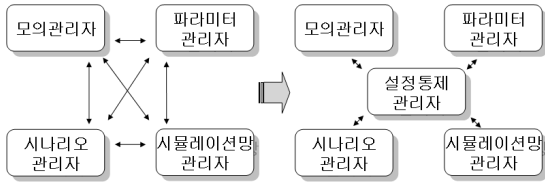


그림 10. 설정통제 관리자 역할

할 수 있는 장점을 제공한다. 그림 10이 이러한 점을 설명하고 있다.

3.7 파라미터 관리자

파라미터 관리자는 각 장비별 초기정보 및 실장비 연동 파라미터의 설정정보를 저장하고 관리한다. INI 형태의 초기 설정정보 파일을 읽고 설정통제 관리자에게 전달하는 역할을 수행한다.

3.8 UI 관리자

UI관리자는 GUI로부터 입력 받은 정보를 설정통제 관리자에게 보내주거나 외부 시뮬레이션으로부터 수신된 각종 정보를 GUI로 보내준다. 장비의 부하를 감소시키기 위해 UI관리자는 TCP/IP 통신을 지원하는 통신 미들웨어를 사용한다. 따라서 GUI는 관측 정보를 생성하는 Backend와 분산 처리 될 수 있다.

앞에서 살펴 본 시뮬레이션 엔진 및 관리자들을 통해 사용자는 분산 시뮬레이션을 포함한 다양한 시뮬레이션 환경을 구축할 수 있다. 또한 기본적으로 제공되는 프레임워크를 상속받아 사용자가 원하는 기능을 쉽게 추가할 수 있다.

4. 프레임워크 활용 결과

4.1 컴포넌트 재사용 현황

현재 시뮬레이션 프레임워크는 총 3개의 사업에 적용되고 있다. 각 사업의 특징은 표 3과 같다.

위의 3가지 사업 중 구현 단계가 완료된 사업 A와 B에 대해서 전체 소프트웨어 중 프레임워크의 사용 비율을 측정하였다. 표 4에서 나타난 것처럼 두 사업의 전체 코드 대비 프레임워크 사용 비율은 시뮬레이션 엔진 11.84%와 관리자 재사용 비율 31.94%를 합하여 43.78%이다.

모의기 별 Framework 재사용 비율은 그림 11, 12와 같다. 프레임워크가 재사용되는 만큼 설계 및 구현 비율을 줄임으로써 비용 절감, 개발 오류 감소 및 개발 기간

표 3. 시뮬레이션 프레임워크 적용 사업

사업명	목적
A	대공유도무기 교전 분석 시뮬레이터
B	실장비 연동 인터페이스 점검용 시뮬레이터
C	대공유도무기 통제 훈련용 시뮬레이터

표 4. 프레임워크 재사용 현황

구분	사업A	사업B	합계
분산장비수	5	10	15
프레임워크 (시뮬레이션엔진)	14.28%	10.91%	11.84%
프레임워크 (관리자)	38.52%	29.42%	31.94%
추가구현	47.2%	59.67%	56.22%

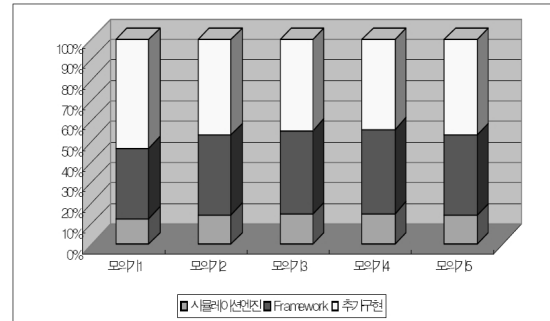


그림 11. 사업A 프레임워크 재사용 비율

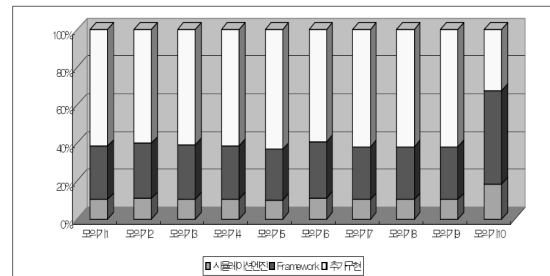


그림 12. 사업B 프레임워크 재사용 비율

단축을 기대할 수 있다.

4.2 분산 및 일체형 시뮬레이터 구성

시뮬레이션 프레임워크로 실제 구현된 분산 시뮬레이션 장비인 운용통제장비의 GUI 화면이 그림 13에 나타나 있다.



그림 13. 분산 시뮬레이션 수행 화면

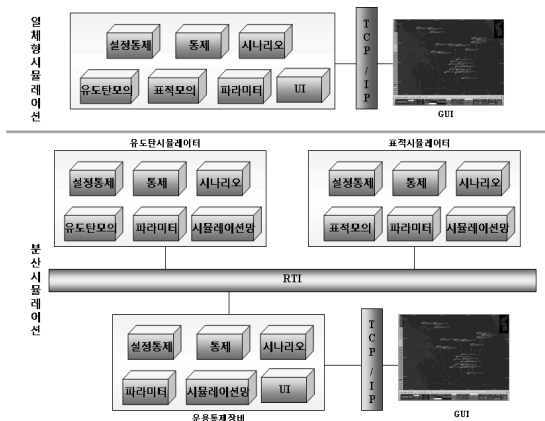


그림 14. 일체형 및 분산 시뮬레이션 구성

현재 시뮬레이션 프레임워크가 적용된 사업의 최종 구현물은 분산 시뮬레이터이지만 이러한 분산 시뮬레이터를 만들기 위해서는 중간에 결과 분석을 위한 평가 및 모델의 효과도 분석을 위해서 일체형 시뮬레이션을 수행할 필요가 있다. 그림 14는 분산 시뮬레이션과 일체형 시뮬레이션을 위한 구성이다. 대공유도무기 효과도 분석을 위해서 일체형 시뮬레이션을 구성한 컴포넌트가 분산 시뮬레이션에 그대로 사용되는 것을 알 수 있다.

5. 결 론

시뮬레이션 프레임워크는 모의기 하부 컴포넌트의 재사용 및 상호운용이 용이한 소프트웨어 구조를 제공한다.

특히 분산 시뮬레이션을 위한 다양한 기능의 컴포넌트를 제공하고 사용자 코드의 추가를 통한 기능 확장을 지원한다. 또한 시뮬레이션 프레임워크의 컴포넌트 재사용성 강화로 일체형 시뮬레이션과 분산 시뮬레이션 간 변경이 용이하므로 개발 단계 별 협업이 효율적으로 이루어질 수 있다. 즉, 제안하는 시뮬레이션 프레임워크는 시뮬레이터를 구성하는데 공통적으로 필요한 기능을 재사용의 단위로 제공하고 사용자가 쉽게 변경하거나 추가할 수 있게 함으로써 사용자의 목적에 따라 유연한 재사용 및 상호운용 방안을 제공할 수 있다. 이를 통하여 시뮬레이터 개발 비용 절감, 개발 기간 단축, 구현용이성 증가 등의 효과를 기대할 수 있고 결과적으로 모의기반획득 환경 구축을 효율적으로 수행할 수 있을 것이다. 제안하는 시뮬레이션 프레임워크의 효과 검증을 위하여 실제 수행중인 사업에 적용하여 재사용 정도를 측정하고 결과를 제시하였다.

현재 사용 중인 프레임워크는 클래스 단위 관리자의 재사용이 가능하다. 이를 플러그인 아키텍처로 변경하기 위한 설계가 진행 중이며 보다 유연한 구조 변경이 가능한 시뮬레이션 프레임워크가 완성될 것이다. 이때 실장비 연동을 위하여 관리자간 통신 최적화 검증도 수행할 계획이다.

참 고 문 헌

1. 박성희, “한국에서의 모의기반획득,” 국방과학기술플러스, Vol. 55, 2008년.
2. 원강연, 최상영, “SBA를 위한 M&S PlugIn-Based Architecture (PBA) 구조에 대한 연구,” 한국군사과학기술학회지, 10(1), 2007년.
3. 이승훈, 손민우, 신동일, 신동규, “SBA (Simulation Based Acquisition)를 위한 통합협업환경 구축에 관한 연구,” 한국컴퓨터종합학술대회 논문집, 35(1), 2008.
4. Doug Buchy, “Joint Modeling and Simulation System,” Joint Modeling and Simulation System Program Office, 2000년 1월.
5. Robert J. Meyer, “Joint Modeling and Simulation System (JMASS)-What it does,... and What it doesn't!”
6. 원강연, 최상영, “M&S PlugIn-Based Architecture Framework 개발,” 정보과학회논문지, 36(2), 2009년.
7. IEEE, “IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Federate Interface Specification.” IEEE Standard No.: 1516.1-2000.
8. Kevin Cox, “A Framework-based Approach to HLA Federate Development,” Simulation Interoperability Workshop, 1998, Fall.

9. 조규태, 이승영, 이한민, 김세환, “실시간 동적 모델 링을 위한 시뮬레이션 엔진 설계,” 제16회 지상무기 학술대회, 2008년.
10. 심준용, 진정훈, 김세환, “M&S Framework를 적용한 효율적인 분산객체 통신모듈 설계,” 한국소프트웨어공학회 학술대회 논문집, 10(1), pp. 208-211, 2008년.
11. Simulation Interoperability Standards Organization Inc., “RPR-FOM Version 1.0 SISO-STD-001.1-1999,” 1999.
12. 심준용, 진정훈, 김세환, “분산객체 통신 시스템의 효율적인 개발을 위한 미들웨어,” 한국정보처리학회 논문집, 14(3), pp. 245-249, 2007년.



조 규 태 (kyutcho@lignex1.com)

2002 숭실대학교 컴퓨터공학부 학사
2004 한국과학기술원 전산학과 석사
2007 한국과학기술원 전산학과 박사수료
2007~현재 LIG넥스원 Maritime연구센터 선임연구원

관심분야 : 모델링&시뮬레이션, 실시간 스케줄링, 지능시스템



심 준 용 (jyshim79@lignex1.com)

2005 우석대학교 컴퓨터공학과 학사
2007 한양대학교 컴퓨터공학과 석사
2007~현재 LIG넥스원 Maritime연구센터 주임연구원

관심분야 : 모델링&시뮬레이션, HLA/RTI, 분산 컴퓨팅



이 용 현 (yhlee80@lignex1.com)

2007 포항공과대학교 전산학과 학사
2007~현재 LIG넥스원 Maritime연구센터 주임연구원

관심분야 : 모델링&시뮬레이션, HLA/RTI, MSDL



이 승 영 (seungyoung.lee@lignex1.com)

2000 인하대학교 전자공학과 학사
2002 인하대학교 전산학과 석사
2002~현재 LIG넥스원 Maritime연구센터 선임연구원

관심분야 : 모델링&시뮬레이션, SBA, T&E



김 세 환 (sachwan.kim@lignex1.com)

1985 경북대학교 전자공학과 학사
1987 경북대학교 전자공학과 석사
2005~현재 LIG넥스원 Maritime연구센터 수석연구원(팀장)
2007~2008 국방과학기술조사 M&S자문위원
2009~현재 국방과학기술수준조사 기술전문가

관심분야 : 모델링&시뮬레이션, SBA, LVC