

플러그인 프레임워크 환경에서의 XML 기반 객체 모델 명세서 검증 도구 설계

유한결*, 이용현*, 심준용*, 김세환*
*LIG넥스원 Maritime연구센터 VM팀
e-mail: youhankyul@lignex1.com

The Design of the Verification Tool of the XML based Object Model Description based on Plug-in Framework

Hankyul You, Yongheon Lee, Jun-Yong Shim, Sae-Hwan Kim
LIG Nex1 Co. LTD. Maritime R&D Center, VM Team

요 약

플러그인 프레임워크는 기존 M&S 프레임워크에서의 문제점인 구성요소 간 높은 종속관계 및 낮은 유연성을 해결하고자 개발된 프레임워크 설계 방식으로, 프레임워크를 구성하는 컴포넌트를 플러그인 형태로 제공하여 구성요소의 추가 및 변경이 동적으로 이루어질 수 있게 한다. 또한 플러그인 컴포넌트 간 통신을 위한 메시지 구조 정의가 실행 시점에 동적으로 결정되어, 컴포넌트 간 상호 독립적인 인터페이스를 보장해준다. 하지만, 플러그인 컴포넌트들의 조합이 이뤄진 상태에서 전체적인 메시지 연결 흐름이 원활하기 위해서는 각 컴포넌트에서 개별적으로 작성된 인터페이스에 대한 검증이 사전에 이뤄져야 한다. 이에 본 논문에서는 XML 기반으로 작성된 객체 모델 명세서에 대한 무결성과 중복성을 검증할 수 있는 객체 모델 명세서 검증 도구에 대해 제안하고 이에 대한 설계내용을 설명하도록 한다.

1. 서론

최근의 분산 시뮬레이션 환경에서의 소프트웨어 개발은 그 규모와 복잡성은 높아지는 반면, 적은 개발비용으로 단 기간 내에 개발할 수 있는 저비용 고효율 환경을 요구하고 있는 실정이다. 이러한 개발추세에 따라, 소프트웨어 개발에 필요한 공통 서비스 환경을 제공하는 프레임워크 개발이 필수적이다.

프레임워크는 특정 기능 군에 속한 응용 소프트웨어 개발에 공통적으로 사용되는 구성 요소와 이들의 아키텍처를 일반화해 부분적으로 구현한 소프트웨어 개발환경으로서, 불필요한 소프트웨어의 중복개발을 피하며 소프트웨어의 재사용성(reusability)을 높여준다. 현재 개발자에게 보다 유연하고 편리한 개발 환경을 제공하기 위해 프레임워크 구조는 계속 발전 중에 있다. M&S 프레임워크[1][2] 구조에서는 소프트웨어 개발에 필요한 공통 기능을 컴포넌트 형태로 생성하고 이를 내부적으로 연동하여 재사용성이 높은 개발환경을 제공해준다. 하지만 컴포넌트 간 상호작용을 위한 메시지 정의가 정적으로 결정됨에 따라 메시지 구조의 유연성이 떨어지게 되는 단점이 있다. 이런 경우, 특정 컴포넌트에서의 간단한 인터페이스 변경이라 할지라도 이와 연관된 다른 컴포넌트들의 소스변경 및 재컴파일 작업이 필요하게 된다. 결국 컴포넌트 간에 결합정도가 높아지며 메시지 통신을 위한 종속관계가 높아지게

되는 문제점이 발생한다.

이를 개선하기 위한 일환으로 컴포넌트간 상호 독립적 인터페이스 및 상호작용이 가능한 플러그인 프레임워크[3]가 개발되었다. 플러그인 프레임워크는 컴포넌트를 DLL 기반의 플러그인으로 제공하여 전체 프레임워크 구성을 동적으로 가져갈 수 있게 해주며, XML 기반의 메시지 객체 모델 및 데이터 기반의 publish-subscribe 방식[7]을 활용하여 메시지 구조의 유연성을 높인다. 또한, 각각의 컴포넌트에서 정의된 XML 기반 객체 모델 명세서를 통해 독립적인 통신 인터페이스를 보장하여, 메시지 기반 통신에 따른 컴포넌트 간의 종속관계를 최소화한다. 하지만, 모든 컴포넌트들 간에 메시지 연결흐름이 원활하게 하기 위해서는, 개별적으로 작성된 모든 객체 모델 명세서에 대한 무결성 및 중복성 여부를 검증해야 할 필요성이 제기된다.

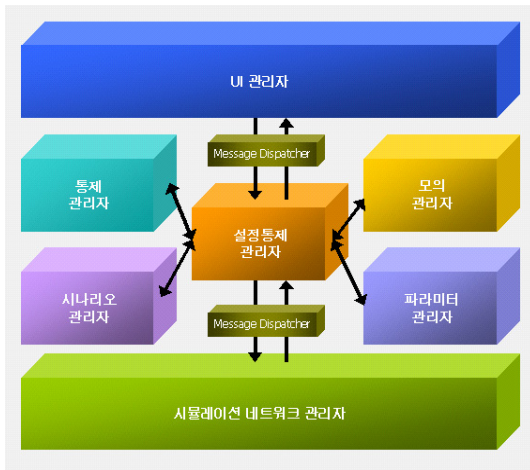
이에 본 논문은 각 플러그인 컴포넌트에서 개별적으로 작성된 XML 기반 객체 모델 명세서들에 대해 검증을 수행하는 XML 기반 객체 모델 명세서 검증 도구에 대한 기본 설계방안을 제안하고, 이를 설명하도록 한다.

2. 관련연구

2.1 M&S 프레임워크

M&S 프레임워크는 시뮬레이션 소프트웨어 개발을 위

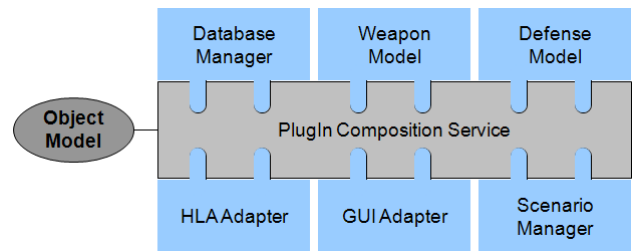
한 공통 구성요소를 컴포넌트 형태로 구현하고 이를 연동하여 소프트웨어 개발을 위한 기반 환경을 제공해준다. 그림 1과 같이 프레임워크를 구성하는 각각의 컴포넌트(관리자)는 내부적으로 메시지 디스패처(Message Dispatcher)를 통해 다른 컴포넌트와 상호작용하여 메시지 기반으로 통신한다. 기본적으로 제공되는 프레임워크 공용 컴포넌트 이외에도 사용자 정의 컴포넌트를 메시지 디스패처와 결합하여 개발할 수 있게 해줌에 따라 개발시 소프트웨어의 재사용성을 높일 수 있다. 하지만 컴포넌트간 통신 메시지 구조 정의가 동적으로 결정되지 못하고 프로그램 소스 코드에 기술되어야 함에 따라, 요구사항 변경에 따른 간단한 메시지 구조 변경이라 할지라도 이와 연관된 모든 컴포넌트들의 리빌드(Re-build) 과정이 필요하게 된다. 이는 메시지 통신에 따른 컴포넌트 간 종속관계가 높으며, 결과적으로 프레임워크의 유연성을 감소시키는 단점을 초래한다.



(그림 1) M&S 프레임워크 구성도

2.2 플러그인 프레임워크

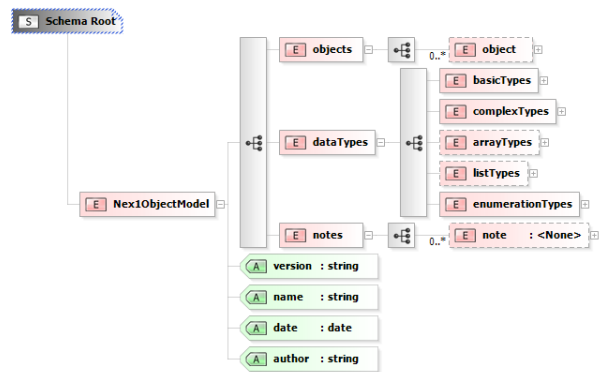
플러그인 프레임워크[4]는 앞 절에서 설명한 M&S 프레임워크에서의 컴포넌트 간 높은 종속관계 및 낮은 유연성의 문제점을 극복하기 위해 플러그인 기반 아키텍처[3]를 적용한 프레임워크이다. 각각의 컴포넌트는 DLL기반의 플러그인 형태로 구현이 되고, Run-time시에 이들이 조합되어 프레임워크가 동적으로 구성이 이뤄진다. 신규 컴포넌트 추가 및 변경에 따른 전체 프레임워크 솔루션의 리빌드(Re-built) 과정이 필요없으며 변경이 필요한 컴포넌트만을 교체해주면 되는 구조상의 유연성을 지니게 된다. 컴포넌트 사이의 종속관계를 최소화하기 위해, 모든 플러그인 컴포넌트는 공통의 추상화된 클래스를 상속받아 구현이 되며, 단일화된 인터페이스로 컴포넌트 간 통신이 이루어진다. 그리고, 컴포넌트 간 독립적인 통신 인터페이스를 제공하기 위해 메시지 구조체 정의가 XML형태로 기술되며, 실행시에 XML파싱(Parsing)을 통해 공통의 추상화된 메시지 객체 인스턴스로 변환되어 단일화된 인터페이스로 메시지 교환이 일어나게 된다.



(그림 2) 플러그인 프레임워크 구성

3. XML기반 객체 모델 명세서 검증도구

이번 장에서는 본 연구의 제안 객체 모델 명세서 검증도구에 대한 기본설계내용을 설명한다. 이에 앞서, 플러그인 프레임워크 구조상에서 컴포넌트 간 상호 독립적 인터페이스 설계를 위한 기반이 되는 객체 모델 구조[7][8]에 대해 간략히 설명하도록 한다. 당사는 컴포넌트간 통신을 위한 객체 모델 구조를 XML로 기술하였으며, 이는 그림 3과 같은 XML 스키마[6]에 의해 정의된다.

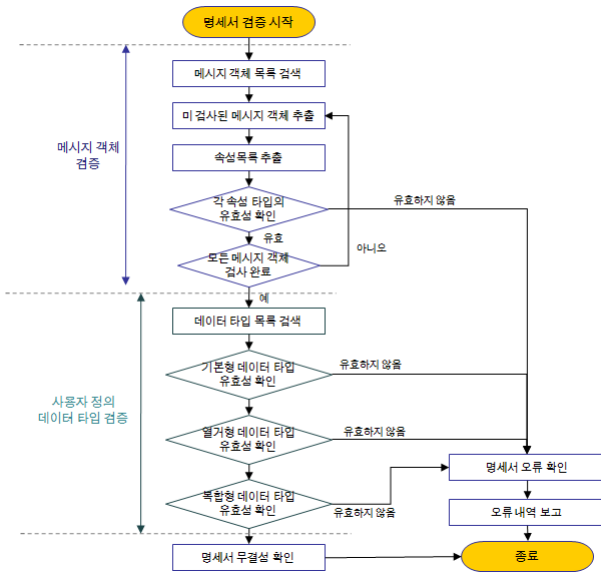


(그림 3) 객체 모델 XML 스키마

객체 모델 구조에서는 컴포넌트 간 송수신하는 데이터의 구조를 정의해주는 메시지 객체와 각종 메시지 객체의 타입을 설정하기 위해 활용되는 데이터 타입으로 구성된다. 메시지 객체는 속성들의 집합으로 구성되며, 각각의 속성에 개별적인 이름과 타입을 지정하게 된다. 데이터 타입은 기본형(Primitive Type: integer, short, double, etc.)과 열거형(Enumeration Type) 및 이들 조합을 통해 구성되는 복합 형식으로 나뉘게 되어 사용자 정의의 데이터 타입을 정의할 수 있다. 위의 구조는 Object Model Template(OMT)[5]서의 Object Class와 Attribute 구조를 참조하여 설계되어 표준 연동모델구조를 지향하고 있다.

이제 위에 설명한 XML 기반의 객체 모델 구조를 통해 작성된 객체 모델 명세서의 무결성 및 중복여부 검증을 수행하는 객체 모델 명세서 검증도구에 대해 설명하도록 한다. 본 객체 모델 명세서 검증도구는 크게 2가지로 분리되어 객체 모델 명세서를 검증한다. 첫째, 신규 혹은 수정된 객체 모델 명세서 자체에 대한 무결성 여부를 검증한다. 그리고, 타 컴포넌트의 객체 모델 명세서와의 비교를 통해 자신의 명세서에서 작성된 객체 모델이 타 명세서에 중복

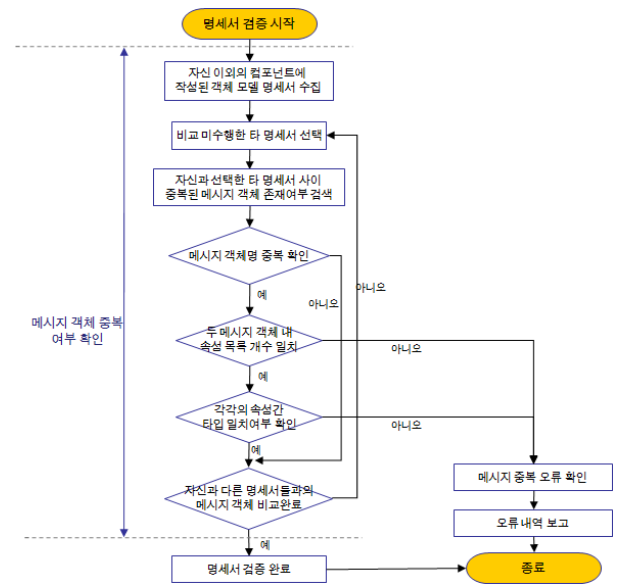
되어 있는지를 확인하여 이에 대한 구조가 일치하는지를 검증할 수 있다.



(그림 4) 신규 객체 모델 명세서 자체 검증 절차

그림 4는 신규 플러그인 컴포넌트에서 작성된 객체 모델 명세서에 대한 자체 무결성 여부를 검증하는 절차를 보여준다. 먼저 컴포넌트 간 통신을 위한 메시지들에 대한 구조를 정의한 메시지 객체가 정확히 기술되었는지를 확인한 후, 각 메시지 객체 속성들의 타입 지정을 위해 활용되는 사용자 정의 데이터 타입의 유효성을 검증한다. 먼저 메시지 객체 검증을 위해, 명세서에서 정의된 메시지 객체 목록을 추출한다. 이후, 각각의 메시지 객체를 구성하는 속성의 타입값이 명세서 내 정의된 데이터 타입 값으로 지정되어 있는지를 확인하여 메시지 객체 검증을 완료한다. 만약, 속성에 지정된 데이터 타입이 정의되지 않는 타입일 경우 개발자에게 오류 내역과 함께 명세서 검증 오류가 발생하였음을 알려준다. 메시지 객체 검증이 완료된 후, 명세서 내 정의된 데이터 타입에 대한 검증을 수행하게 된다. 일차적으로, 명세서 내 사용자 정의 데이터 타입 목록을 가져온다. 그 후, 기본형 및 열거형 타입으로 정의된 각각 데이터 타입 목록에 대해 각각의 값이 유효한지를 확인한다. 이후 이들의 조합으로 구성되는 복합형식의 데이터 타입 목록에 대해 검증을 수행하며, 각 복합형식 데이터 타입을 구성하는 각 필드의 데이터 타입이 정의된 데이터 타입으로 설정되었는지를 확인하여 유효성 여부를 판단하게 된다.

위에 설명한 절차를 수행하여 신규 객체 모델 명세서의 무결성이 검증되면, 기존 컴포넌트들에서 작성된 객체 모델 명세서와 비교하여 메시지 객체의 중복여부를 판단하게 된다.



(그림 5) 타 객체 모델 명세서와의 비교를 통한 중복 메시지 충돌여부 확인

그림 5는 신규 혹은 변경된 객체 모델 명세서를 기존의 다른 객체 모델 명세서와 비교하여 중복 메시지에 대한 구조일치 여부를 검증하는 절차를 보여준다. 먼저 기존 컴포넌트들의 객체 모델 명세서를 수집하여 전체 객체 모델 명세서를 비교할 수 있도록 기초작업을 수행한다. 이후, 수집된 모든 객체 모델 명세서를 새로 작성된 객체 모델 명세서와 비교하여 중복된 메시지 객체가 있는지를 검색한다. 메시지 객체의 중복여부는 메시지 객체 명을 비교하여 확인하고, 동일한 객체 명이 있는 경우를 메시지 객체 중복이라 간주한다. 만일 두 명세서 사이에 중복되는 메시지 객체가 있을 경우, 두 메시지 객체의 구조가 일치하는지를 확인하여 충돌여부를 알아낸다. 중복 메시지에 대한 충돌여부를 확인하기 위해 다음과 같이 2가지 절차를 수행한다. 먼저 메시지 객체를 구성하는 각각의 속성들의 개수를 파악하고 이 둘의 개수가 일치하는지를 확인한다. 그리고 이 두 속성집합에 대하여 순서대로 각각의 속성 타입을 비교하여 타입이 동일한지를 확인하여 두 메시지 객체의 구조가 동일한지를 검증하게 된다. 만약 이 두 가지 조건이 모두 만족하지 못할 경우, 개발자에게 메시지 충돌 오류가 발생하였음을 알려주게 된다.

본 검증도구에서 제공하는 위의 2가지 검증절차를 모두 수행하면 신규 객체 모델 명세서의 검증이 완료되며, 이를 적용한 플러그인 컴포넌트는 다른 컴포넌트와 아무런 충돌없이 데이터 교환이 이뤄지게 된다.

4. 결론 및 향후 연구방안

본 논문은 플러그인 프레임워크 환경에서 각 컴포넌트에서 작성된 XML 기반 메시지 구조 정의 명세서들에 대해 검증을 수행하는 객체 모델 명세서 검증 도구에 대한 기본설계 방안에 대해 제안하였다. 이를 통해 플러그인 컴

포넌트의 독립적인 인터페이스 보장을 위해 작성된 사용자 정의 객체 모델 명세서의 무결성을 사전에 검증할 수 있으며, 구성요소 간 조합이 이뤄졌을 때 발생할 수 있는 메시지 중복 및 충돌등의 예외적인 사항을 사전에 방지할 수 있게 된다.

향후, 본 논문에서 제시한 기본 설계 내용을 구체화시켜 구현하고 이를 실제 소프트웨어 개발에 적용함으로써, 제시한 검증 도구에 대한 실효성을 확인하고자 한다. 또한 현재의 동적 인터페이스 설계 방식에서 추가적으로 발생할 수 있는 메시지 구조 정의 관련 예외사항을 계속적으로 찾아내어 객체 모델 명세서 검증 도구에 반영하도록 할 예정이다.

참고문헌

- [1] 이용현, 이승영, 정하민, 김세환, “분산시물레이션 환경의 운용통제 프레임워크 설계”, 소프트웨어공학 합동워크샵, 2008.07
- [2] 심준용, “M&S Framework를 적용한 효율적인 분산객체 통신모듈 설계”, 한국소프트웨어공학 학술대회 논문집 제10권 1호, 2008
- [3] 원강연, 최상영, “M&S PlugIn-Based Architecture Framework 개발”, 한국정보과학회논문지, 시스템 및 이론 제36권 제2호(2009.4)
- [4] 심준용, 이용현, 김세환, “컴포넌트 재사용을 위한 DLL 플러그인 프레임워크 설계”, 한국정보처리학회 춘계학술대회 논문집 제 17권 2호, 2010.11
- [5] <http://www.msco.mil/HLAComplianceTesting.html>
- [6] <http://www.w3.org/XML/Schema>
- [7] 이용현, 심준용, 조규태, 이승영, 김세환, “Publish-Subscribe 메시지 프로토콜을 위한 XML 기반의 Object Model 설계”, 한국시물레이션학회 춘계학술대회, 2010.05
- [8] 이용현, 심준용, 김세환, “사용자 정의 형식을 지원하는 XML 기반 객체 모델의 구조 설계”, 한국정보처리학회 논문집 제 17권 2호, 2010.11