

DLL기반 소프트웨어 통합을 위한 버전관리 방법

김민아, 심준용, 위성혁
LIG넥스원 Maritime 연구소
e-mail: minah.kim@lignex1.com

A Method of Version Check for DLL based Software Integration

Min Ah, Kim, Junyong, Shim, Soung Huouk, Wi
Maritime R&D Center, LIG Nex1 Co.,Ltd.

요 약

소프트웨어 개발의 재사용성과 신뢰성을 높이기 위해서 DLL 기반의 소프트웨어로 구성된 M&S 프레임워크를 개발하였다. M&S 프레임워크를 적용한 소프트웨어는 짧은 기간에 개발하여 비용을 절감할 수 있는 장점은 있으나, 프레임워크에서 사용된 다수 DLL의 버전관리의 어려움이 있다. 본 논문에서는 DLL 기반 소프트웨어 통합을 위한 버전관리 방법을 제시한다.

Keywords : M&S 프레임워크, DLL 버전관리, 통제관리자, 분산시물레이션

1. 서론

국방 소프트웨어 분야의 M&S (Modeling & Simulation) 수요가 증가함에 따라 개발기간 단축과 비용 절감의 방법으로 재사용 가능한 M&S 프레임워크를 개발하였다. M&S 프레임워크는 시물레이션 소프트웨어 개발에 필요한 공통 기능을 컴포넌트 형태로 생성 및 연동함으로써 재사용가능한 개발환경을 제공한다.

초기 M&S 프레임워크의 구현방식은 컴포넌트 간 메시지 통신 구조가 코드 수준에서 결정된다. 즉, 연관된 컴포넌트의 메시지 추가 또는 수정이 발생하면 해당 컴포넌트의 재컴파일 과정이 필요하다. 따라서 메시지 통신에 따른 컴포넌트 간 종속성은 증가하고, 프레임워크 구성에 대한 변경 용이성은 감소한다[1].

이런 문제를 해결하기 위해 개선된 M&S 프레임워크는 초기 요구기능에 따라 컴포넌트를 동적으로 재구성할 수 있는 Dynamic Linked Library (이하 DLL) 기반의 플러그인 구조로 설계하여[2], 메시지 구조의 유연성과 독립성을 높이기 위한 데이터 기반의 publish-subscribe (이하 PS) 방식[3]을 활용하여 컴포넌트 간 종속성을 최소화 하였다.

개선된 M&S 프레임워크를 적용한 소프트웨어는 요구사항 추가, 변경이 간편한 장점이 있으나, 소프트웨어 개발 시 각각의 컴포넌트가 DLL 단위로 구성되어, 통합하는 과정에서 다수의 DLL의 버전 관리에 대한 어려움이 발생한다[4].

본 논문의 2장에서는 기존 M&S 프레임워크의 구조와 개선된 M&S 프레임워크 구조를 소개하고 문제점을 기술한다. 3장은 문제점 해결을 위해 M&S 프레임워크기반으

로 개발된 컴포넌트 DLL 의 버전관리방법과 컴포넌트로 구성된 모의소프트웨어의 구성 DLL 버전 확인 방법, 모의 소프트웨어가 참여한 분산시물레이션환경에서의 DLL 버전 관리 방법을 기술하고 4장에서 결론을 맺고 마친다.

2. M&S 프레임워크 구조

2.1 기존 프레임워크 구조

분산시물레이션환경의 모의 소프트웨어 개발 시 재사용을 목적으로 개발된 M&S 프레임워크는 그림 1과 같이 7 종의 관리자와 5 종의 라이브러리를 기본으로 사용자 정의에 따라 확장 가능한 구조를 갖는다.

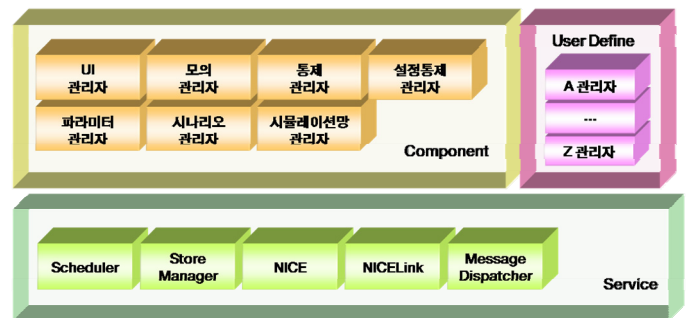


그림 1 기존 M&S 프레임워크 구조

컴포넌트 계층의 관리자는 분산시물레이션환경에서 요구되는 공통 기능[5]으로 모의 소프트웨어 구성 시 수정 없이 공통 모듈로 사용하고, 서비스 계층의 5 종의 라이브러리(스레드 관리, 로그 저장, TCP, UDP, RTI 통신 지원,

내부 컴포넌트 간 메시지 교환 처리)는 컴포넌트 레이어의 관리자 개발 시 사용되는 서비스로 제공한다.

기존 M&S 프레임워크는 제공된 7종의 관리자 외에도 사용자 정의 컴포넌트를 Message Dispatcher (이하 MD)를 사용하여 기능을 확장하고, 소프트웨어를 재사용 할 수 있다. 하지만, MD의 구조[2]적인 문제로 컴포넌트 간 종속성이 높아져, 유지보수 시 유연성이 떨어지는 단점이 있다.

2.2 개선된 프레임워크 구조

기존 프레임워크의 단점인 컴포넌트 간 종속성을 해결하기 위한 그림 2와 같이 프레임워크 구조를 개선하였다.

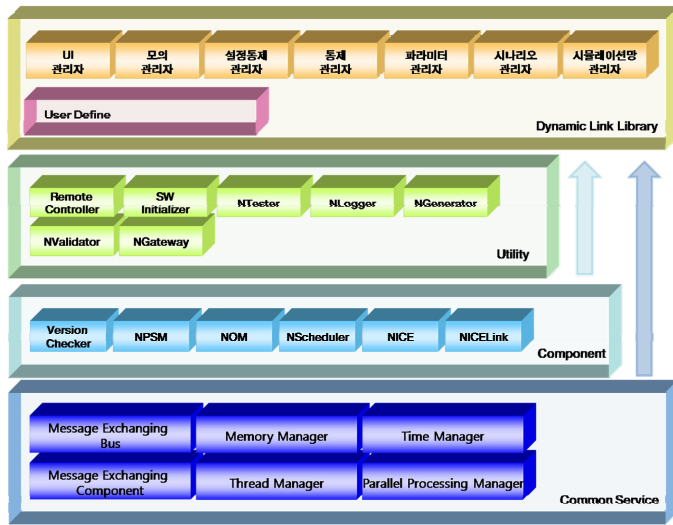


그림 2 개선된 M&S 프레임워크 구조

기존 프레임워크와의 차이는 컴포넌트 간 메시지 교환 처리 방식인 MD를 사용하지 않고, 메시지 구조의 유연성을 제공하는 데이터 기반의 PS 방식[3]을 사용하여 기능처리부와 인터페이스부를 분리하여 컴포넌트 간 종속성을 최소화한 것이다. 이를 위해 개선된 프레임워크의 4계층 중 component 계층과 common Service 계층은 PS 구현을 위해 필수적인 NOM[7], NPSM[2], Message Exchanging Component[2], Message Exchanging Bus[2] 컴포넌트를 제공한다.

그리고 개선된 프레임워크는 요구사항 추가 또는 수정 시 유연하게 변경하기 위하여 요구사항에 따라 동적 플러그인이 가능한 DLL 형태로 관리자를 제공하고, Dynamic Link Library 계층으로 관리한다.

utility 계층의 경우 프레임워크 기반의 소프트웨어 개발 시 직접적으로 사용되는 라이브러리는 아니지만, 선택적으로 사용할 수 있는 별도의 어플리케이션 모음이다.

개선된 프레임워크는 DLL 단위의 관리자를 통합하는 과정이 필요하고, 동적으로 플러그인 되는 DLL의 버전이 검증된 버전인지 여부를 확인할 수 있어야 한다.

2.3 문제점 및 해결방안

개선된 M&S 프레임워크를 사용하면 기존 M&S 프레임워크의 기능을 효율적으로 재사용 할 수 있다. 하지만, 프레임워크 기반의 소프트웨어 구동 시 다수의 DLL이 플러그인 되는 과정에서 DLL 버전이 관리되지 않아, 검증된 DLL이 아닌 버전의 DLL이 플러그인 되어 이상 동작하는 문제가 발생했다. 이런 문제를 해결하기 위해 플러그인 된 모든 DLL의 크기와 컴파일 날짜를 비교하여 검증된 버전인지 여부를 확인하였다.

이런 문제는 앞으로도 지속적으로 발생할 수 있으므로, 개선된 프레임워크에 추가적인 필수 컴포넌트를 추가하여 DLL 생성단계부터 버전을 관리할 수 있는 방법이 필요하다. 그림 3은 이런 문제를 해결하기 위한 버전 관리 방법으로 DLL 생성 단계부터 소프트웨어 통합, 분산시물레이션에 적용되는 3 단계의 DLL 버전관리 단계이다.

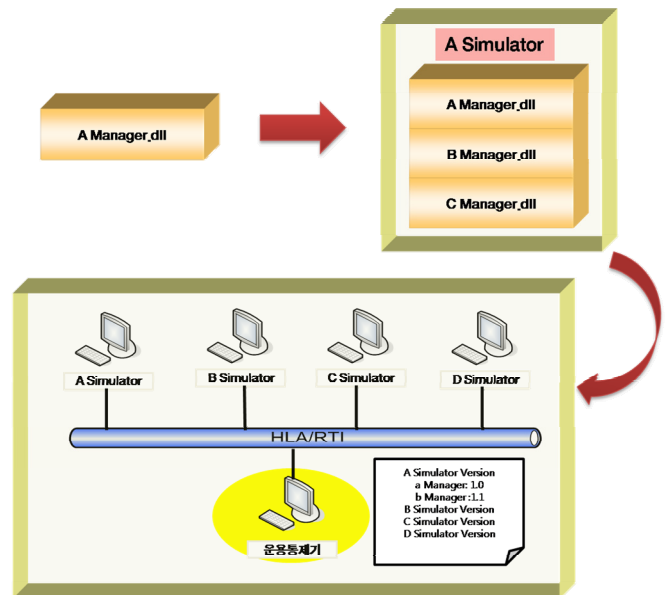


그림 3 DLL 버전관리 단계

3. DLL 버전관리 방법

3.1 관리자 DLL 버전관리 방법

첫 번째 단계는 모의 소프트웨어를 구성하는 관리자 DLL 생성 및 빌드 시 버전 관리이다. 개선된 프레임워크에는 DLL 생성 및 빌드 시 버전을 자동으로 갱신해주는 Version Checker 컴포넌트를 제공한다.

Version Checker는 버전을 갱신할 수 있는 SetVersion 함수와 최신 버전정보를 얻을 수 있는 GetVersion 함수를 제공한다.

Version Checker를 사용하여 컴포넌트 DLL 버전 관리 방법은 그림 4와 같다.

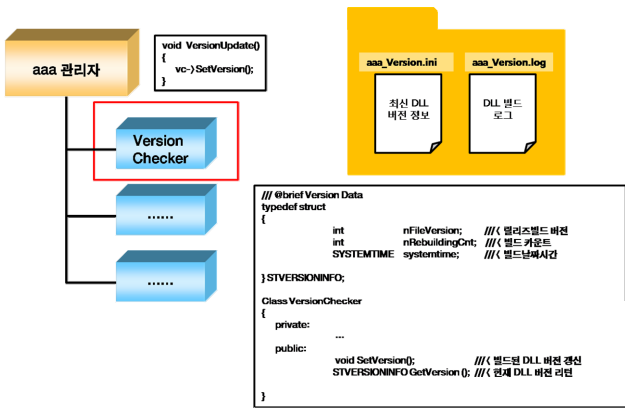


그림 4 Version Checker를 사용한 컴포넌트 DLL 버전관리

관리자 DLL에서 Version Checker의 객체를 생성하여 SetVersion함수를 호출하면, SetVersion함수 내부에서 관리자 컴포넌트 빌드 시 생성된 최종 DLL 버전정보를 읽어와 날짜와 빌드 횟수를 갱신하여 버전정보를 업데이트 한다. 관리자 DLL 개발 시 Version Checker를 사용하면 유지보수 중 리빌드의 버전이 자동으로 갱신되어 관리된다.

3.2 모의 소프트웨어의 DLL 버전관리 방법

두 번째 단계는 모의 소프트웨어 구성 시 플러그인 되는 관리자 DLL의 버전 관리 방법이다. 모의 소프트웨어는 관리자 DLL의 플러그인으로 구성되며, 프레임워크에서 제공하는 관리자 DLL과 사용자가 정의한 관리자 DLL의 조합으로 구성된다.

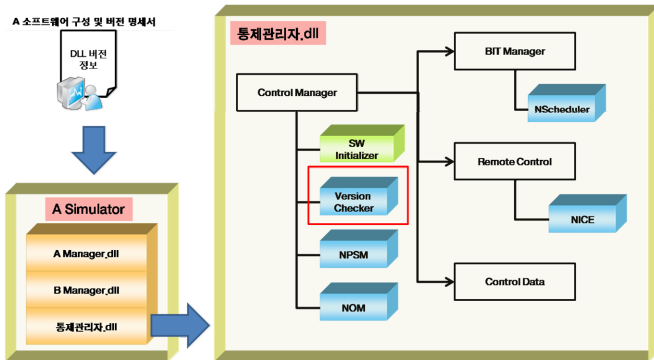


그림 5 모의소프트웨어 DLL 버전 관리 절차

모의 소프트웨어를 구성하기 위한 방법은 그림 5와 같다. 우선, 모의 소프트웨어 개발자 (DLL Integrator)는 모의 소프트웨어에 플러그인 되는 관리자 DLL의 구성과 각 DLL의 최신 버전을 명시한 명세서를 작성한다. 개발자는 관리자 DLL의 버전 작성 시 서버에 저장된 최신 버전을 확인해야 한다. 명세서 작성이 완료되면 모의 소프트웨어를 구동하고, 모의 소프트웨어를 구성하는 DLL의 버전을 Version Checker의 GetVersion 함수를 호출하여 리턴받아 개발자가 작성한 명세서의 버전과 비교하여 플러그인 된

DLL의 버전이 최신인지 여부를 확인한다.

프레임워크에는 이런 기능을 제공하기 위해서 Dynamic Link 계층에 통제관리자를 제공한다. 기존 프레임워크의 통제관리자는 분산 시뮬레이션 수행 시 원격 모의기의 상태를 감시하고 제어하는 기능을 수행하였으나, 개선된 M&S 프레임워크의 통제관리자는 기존의 통제기능[6]에 Version Checker를 사용한 버전 확인 기능을 추가하여 그림 5의 구조로 설계한다.

프레임워크 기반의 모의 소프트웨어는 동적으로 DLL이 구성되기 때문에 프레임워크에서 제공하는 통제관리자를 사용하면 소프트웨어 구동 시 플러그인 되는 DLL의 버전이 항상 최신인지 여부를 확인 할 수 있다.

3.3 분산 시뮬레이션의 DLL 버전관리 방법

세 번째 단계는 분산 시뮬레이션 환경에 연동되는 모의 소프트웨어의 DLL 버전 관리 방법이다. 첫 번째와 두 번째 단계를 통해 완성된 각각의 모의 소프트웨어는 그림 6과 같이 분산시뮬레이션에 참여한다.

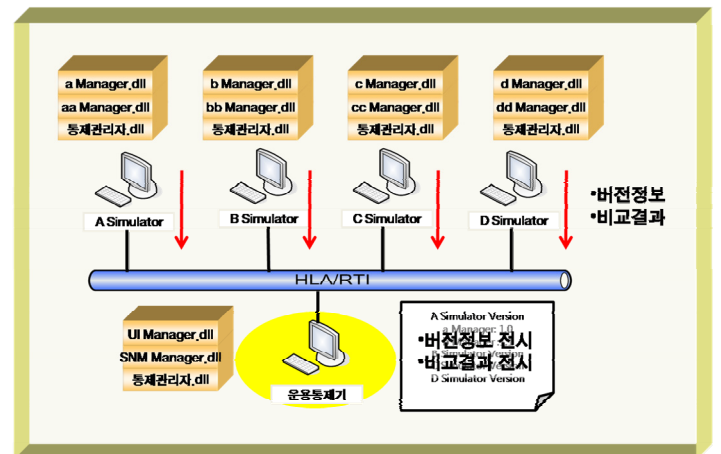


그림 6 원격 소프트웨어의 버전확인 방법

모의 소프트웨어가 정상 구동되어 시뮬레이션 망에 연동되면 각 연동 모의기는 상태정보를 전송하기 전, 두 번째 단계의 DLL 버전확인 결과와 모의 소프트웨어의 구성 DLL 버전을 운영통제기로 전송하여 운영자가 모의기의 버전을 확인할 수 있도록 전시한다. 운영자는 현재 구동되고 있는 모든 모의 소프트웨어의 구성 DLL 버전을 확인하여, 공용 DLL의 버전이 상이한 모의 소프트웨어를 식별할 수 있다.

분산 시뮬레이션 수행 시 연동하는 각각의 모의 소프트웨어의 공용 DLL 버전이 상이할 경우 시뮬레이션 결과가 기대 값과 다르거나, 예상치 못한 오류가 발생할 수 있으므로, 시뮬레이션 준비 과정에서 DLL의 버전을 확인하여 잠재된 문제를 해결하고, 신뢰성 높은 시뮬레이션을 수행할 수 있다.

4. 결론

국방 M&S를 적용한 시뮬레이션 기반 무기체계 획득(SBA: Simulation Based Acquisition)과 설계(SBD: Simulation Based Design)에 대한 관심이 높아짐에 따라 시뮬레이션을 위한 소프트웨어 개발의 수요가 증가하고 있다. 이런 다양한 소프트웨어 수요를 단시간에 충족하여 저비용 고효율의 방법으로 M&S 프레임워크를 개발하였다. M&S 프레임워크는 동적구성이 가능한 DLL 기반의 플러그인 구조로 컴포넌트 간 독립성을 보장하고, 모의 소프트웨어 개발 시 DLL 간 유연한 재구성이 가능하다.

M&S 프레임워크는 컴포넌트가 DLL 형태로 구현되기 때문에 각 컴포넌트의 기능 추가와 수정에 따른 DLL 버전을 관리하는 기능이 필수적이다. 본 논문에서는 프레임워크의 DLL 버전을 자동으로 갱신하는 Version Checker와 소프트웨어 개발 시 DLL 통합과정에서 사용자가 명시한 버전과 통합된 DLL 버전을 비교하여 확인 하는 통제관리자를 소개하고, 분산 시뮬레이션 수행 시 다수의 원격 소프트웨어의 구성 DLL을 통제컴퓨터에서 확인하는 방법을 제시하였다. M&S 프레임워크를 기반으로 개발된 소프트웨어에 이런 버전관리 방법을 적용하면 버전이 상이하여 발생할 수 있는 잠재된 문제를 해결하고, 신뢰성 높은 시뮬레이션을 수행할 수 있다.

향후 DLL 기반의 소프트웨어 통합 시 DLL의 버전확인만으로 호환 가능성 여부를 판단할 수 있는 DLL 버전 규칙의 표준화 방안에 대한 연구가 필요하다.

참고문헌

- [1] 유한결, 이용현, 심준용, 김세환, “플러그인 프레임워크 환경에서의 XML 기반 객체 모델 명세서 검증 도구 설계”, 제 35회 한국정보처리학회 춘계학술발표대회 논문집 제 18권 제1호 (2011.05)
- [2] 심준용, 이용현, 김세환 “컴포넌트 재사용을 위한 DLL 플러그인 프레임워크 설계” 제34회 한국정보처리학회 추계학술대회 논문집 제 17권 2호 (2010.11)
- [3] 이용현, 심준용, 조규태, 이승영, 김세환, “Publish-Subscribe 메시지 프로토콜을 위한 XML 기반의 Object Model 설계”, 한국시뮬레이션학회 춘계학술대회 (2010.05)
- [4] <http://ko.wikipedia.org>
- [5] 조규태, 심준용, 이용현, 이승영, 김세환 “모의기반획득을 위한 시뮬레이션 아키텍처 프레임워크 개발”, 한국시뮬레이션학회 논문지 Vol.19, No.3, (2010.09)
- [6] 김민아, 이승영, 김세환 “실 무기체계 연동시뮬레이션 통제를 위한 소프트웨어 구조연구”, 국방과학연구소 창설 40주년 기념 종합학술대회 (2010.08)
- [7] 이용현, 심준용, 김세환 “사용자 정의 혁식을 지원하는 XML 기반 객체 모델의 구조 설계”, 제34회 한국정보처리학회 추계학술대회 논문집 제 17권 2호 (2010.11)