

유럽 우주비행체 시뮬레이터 연구개발 동향

이훈희*

Trend of European Spacecraft Simulator Development

Lee, Hoon-hee*

ABSTRACT

As the cost of the operational spacecraft simulator is a large portion of the spacecraft cost and its requirements increase, it is given the pressures to reduce the cost and the development period. For that reason, the space industry with the coordination of ESA has made a study of the reuse in a wide scope across space projects. Its scope includes not only simulation software itself but also software requirements, design, work experience and developing process/methodology. The standard for simulator model portability and the efficiency enhancement of the development process as a result have been successfully applied to several projects. This paper describes the recent trend of the spacecraft simulator in Europe and the present status of its the study and the development.

초 록

유럽의 우주비행체 시뮬레이터는 요구사항의 증가와 함께 우주비행체 개발 비용에서 높은 비율을 차지하여 개발 비용 절감과 개발 기간 단축의 압력에 직면하고 있다. 현재 유럽은 유럽우주국을 중심으로 다양한 우주 프로젝트 경험을 이용하여 소프트웨어 자체의 재사용 연구뿐만 아니라 소프트웨어 요구사항, 설계 노하우, 개발방법론, 경험 등 우주 프로젝트 간의 광범위한 자원 재사용 연구를 활발히 진행하고 있다. 재사용 연구 결과는 시뮬레이션 모델의 표준 제정과 시뮬레이터 개발 프로세스의 효율성 증대로 나타나고 있으며 후속 프로젝트에 적용되어 효과를 입증하고 있다. 본 논문에서는 최근 유럽의 우주비행체 시뮬레이터의 연구동향 추세와 시뮬레이터 소프트웨어 개발 동향을 기술한다.

Key Words : Operational Simulator(운영시뮬레이터), Trend(동향), Simulation(시뮬레이션), Spacecraft(우주비행체), Satellite(인공위성), Standard(표준), Software(소프트웨어)

* 이훈희, 한국항공우주연구원 통신해양기상위성사업단 통해기체계팀
lhh@kari.re.kr

1. 서론

1.1 우주비행체 시뮬레이터 개요

하나의 우주프로젝트 기간 동안 사용자의 요구나 목적에 따라 다양한 시뮬레이터가 개발된다. 유럽에서 통상적으로 “우주비행체 동역학 시뮬레이터(Dynamic Spacecraft Simulator)”, “우주비행체 시뮬레이터(Spacecraft Simulator)”, “우주비행체 운용 시뮬레이터(Spacecraft Operational/Operations Simulator)”, “우주비행체 시스템 시뮬레이터(Spacecraft System Simulator)” 등으로 불리는 시뮬레이터는 우주환경과 함께 전체 우주비행체의 완전한 동작을 모사하여 운용 절차의 검증, 지상관제 시스템의 시험, 운영자의 교육 및 훈련 등을 목적으로 하는 최종 단계의 시뮬레이터를 의미한다. 본 논문에서는 이러한 우주비행체 시뮬레이터(이하 시뮬레이터)에 대해서 중점적으로 설명하며 그림1은 TM(Telemetry)/TC(Telecommand) 교환을 위한 시뮬레이터 소프트웨어와 지상관제국과의 관계를 나타내고 있다. 실제로 그림의 시뮬레이터 소프트웨어는 지상관제소프트웨어의 지원 목적의 한 부분으로 구성된다. 국내에서도 이미 무궁화위성, 아리랑위성, 통신해양기상위성 등의 시뮬레이터를 개발 혹은 활용하거나 현재 이용 중이다.

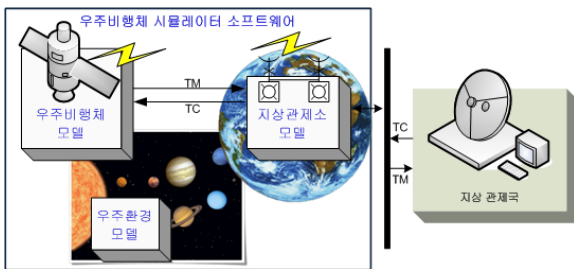


그림 1. 우주비행체 시뮬레이터와 실제 지상관제국

1.2 시뮬레이터 소프트웨어 분류

일반적으로 유럽의 시뮬레이터 소프트웨어는 그림 2와 같은 구조를 이루고 있다. ‘Windows’나 ‘Linux’ 등과 같은 최하위의 운영체제 위에는 분산객체 간의 통신을 위한 미들웨어(Middleware)가 위치하기도 하며 그 상 위에는 순수한 시뮬레이션 소프트웨어가 위치한다. 이것은

크게 실제 장치나 환경 등을 모사하는 시뮬레이션 모델과 에뮬레이터(Emulator) 부분과 이를 지원하는 시뮬레이션 기반소프트웨어 부분으로 나눌 수 있다. 통상적으로 “시뮬레이션 기반소프트웨어”라고 하면 다양한 종류의 시뮬레이터에 공통적으로 사용될 수 있는 비종속적 범용 소프트웨어를 의미한다. 보통 기능적으로 시뮬레이션 모델을 실행/제어/감시를 하기 위한 범용 시뮬레이션 개발/실행/분석 환경 혹은 플랫폼을 의미하며 그림 2의 노란색과 회색 블록과 같이 시뮬레이션 커널과 시뮬레이션 제어/관리, 기타 사용자를 위한 유용한 소프트웨어로 구성된다.

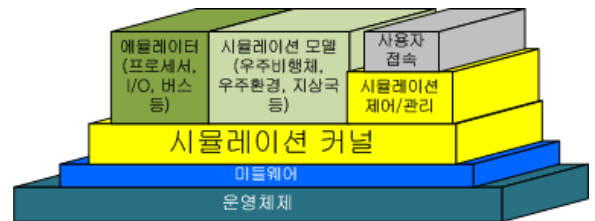


그림 2. 시뮬레이터 소프트웨어 구성

최근에는 프로젝트에 대한 종속성이 적은 범용 에뮬레이터나 시뮬레이션 모델의 경우도 시뮬레이션 기반 소프트웨어에 포함시키고 있다. 이러한 다양한 시뮬레이션 기반소프트웨어를 이용해서 시뮬레이터 개발 비용을 절감하고 있다.

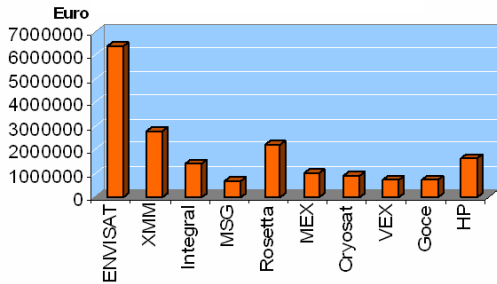
본 논문에서는 유럽의 시뮬레이터 연구/개발 동향을 살펴본다. 2장에서 최근 유럽의 시뮬레이터의 연구/개발을 촉진하는 주요 요구사항과 현재 문제점 혹은 난점에 대해 알아보고 이에 대한 유럽 우주 기관 및 기업들의 대응전략을 기술한다. 3장에서는 유럽의 우주기업들이 최근 기술을 적용하여 개발한 주요 시뮬레이터 기반소프트웨어에 대한 특징과 변화를 살펴봄으로써 시뮬레이터 개발동향을 파악한다. 4장 결론에서는 유럽의 시뮬레이터 연구/개발의 미래를 예측하여 본다.

2. 시뮬레이터 연구동향

2.1 요구사항 추이

유럽의 시뮬레이터 개발/공급 업체들은 캐나다, 미국, 아시아, 중동 등의 다양한 사용자의 요구에 부응하는

시뮬레이터 시스템을 공급하며 국제적인 시장 경쟁력을 확보/유지하고 있다. 최근 중국이나 인도와 같은 후발 우주시장 진출자들이 가격경쟁력을 앞세워 시장점유율을 높이는 가운데 기존 시장의 높은 점유율을 자랑하던 기업들은 기존의 우위를 유지하기 위한 노력을 하고 있다.



자료 : ESA OPS-G Forum, 2008
그림 3. 유럽 우주비행체 시뮬레이터 개발 비용

한편 시뮬레이터 개발 연구 정책의 중심이 되는 유럽 우주국(ESA : European Space Agency)의 기금에 경제적 압력이 작용하는 가운데 우주비행체 개발비용의 약 1% 가량을 차지하는 우주비행체 시뮬레이터는 개발비용 절감에 대상이 되고 있는 게 사실이다. 또한 우주비행체 개발주기의 단축화 추세에 따라 시뮬레이터의 개발시간도 종속적으로 단축 압력에 직면하고 있다. 게다가 현재 통신해양기상위성의 사례와 같이 임무를 위한 많은 수의 다기능 탑재체를 장착하는 추세 속에 우주비행체 시스템의 복잡도가 높아지면서 시뮬레이터의 요구사항은 증가하고 있다. 이와 같이 시뮬레이터의 개발 비용 감소 및 기간 단축의 압력과 더불어 사용자 요구사항은 한층 더 높아지고 있다.

2.2 문제점 및 극복과제

유럽우주국의 ESOC(European Space Operations Centre)이나 ESTEC(European Space Research and Technology Centre)은 우주프로젝트에서 수많은 시뮬레이터 소프트웨어를 개발하면서 기술적 개선 및 비용절감을 위한 시뮬레이터 소프트웨어의 생산성 향상을 위한 연구를 했다. 또한 Astrium, VEGA, Science Systems 등의 우주기업들이 과학실험용, 상용서비스용 시뮬레이터 제작을 통해 개발 과정에서

나타나는 문제를 해결하면서 개발 기술을 축적해 왔다. 그러나 새롭게 나타나는 문제나 우주비행체 개발과의 종속성에 의한 고질적인 개발 난제에 여전히 노출되어 있다. 표 1은 최근에 우주비행체 개발과정에서 발생하는 문제점이나 변동사항이 시뮬레이터 개발에 어떠한 영향을 미치는지를 나타내고 있다.

표 1. 우주비행체 개발이 시뮬레이터에 미치는 영향

우주비행체 관련 항목	우주비행체 변동사항/문제점	시뮬레이터에 파급 영향
탑재 컴퓨터장치	고성능 프로세서 사용, 주변 하드웨어 복잡, 표준화 미흡	에플레이터 및 주변 장치 모델링 복잡성 증가/시뮬레이터 하드웨어 사양 증가
탑재 소프트웨어/데이터베이스	탑재소프트웨어 규모와 TC/TM 및 파라미터 증가	시뮬레이션 내부 변수 및 파라미터 양 증가
데이터처리 시스템/임무탑재체	고성능화, 자율 기능의 증가, 동시 탑재수 증가	모델링의 복잡성 증가/시뮬레이터 하드웨어 사양 증가
개발 일정	사용자 요구사항 전달 지연	시뮬레이터 개발 기간 단축 압력
	탑재소프트웨어/데이터베이스의 오류	시뮬레이션 모델 검증 시간 증가
	탑재소프트웨어/데이터베이스의 전달 지연	시뮬레이션 모델 검증 시간 증가
	우주비행체 개발 기간의 단축	시뮬레이터 개발 기간 단축 압력
	임무탑재체의 설계 확정의 지연	임무탑재체 모델 개발 시간 단축
	잡은 사용자 요구 사항 변경	재검증 비용 소비
	기술정보 및 설계 정보의 전달 지연	시뮬레이션 모델 설계 시간 단축
	개발 예산 축소	시뮬레이터 개발 예산 축소

표 1에 기술된 바와 같이 시뮬레이터 개발이 시뮬레이터의 개발이 우주비행체 개발과 관련된 여러 변동사항으로부터 직접 영향을 받는 종속적 문제점을 내포한다. 주요한 내용을 설명하면 다음과 같다.

• 탑재컴퓨터장치 모델의 개발 제약

내부의 고속 통신 용 칩이나 프로세서 등은 독립된 애플레이터 형태로 개발하여 시뮬레이터에서 사용된다. 애플레이터를 사용하는 이유는 실제 탑재소프트웨어 이미지를 사용하여 데이터처리시스템, 자세제어시스템의 정확한 타이밍의 동작을 모사할 수 있기 때문이다. 또한 입출력포트와 메모리에 직접적인 조작용을 가할 수 있어 다양한 고장시험과 가속화된 시간에서 고속 시험이 가능하다는 장점이 있다. 특히 실제 우주비행체에 새로운 탑재 소프트웨어 패치를 전송하기 전의 사전검증시험을 위한 효율도가 매우 높다. 결국 우주비행체 상위레벨의 모델링 충실도를 완전하게 제공해 줄 수 있어 애플레이터 연구 개발은 앞으로 계속 이어질 전망이다.

최근에 탑재컴퓨터장치는 프로세서의 고성능화, 주변 회로의 복잡성 증가와 잦은 재구성으로 인하여 시뮬레이터 개발에 영향을 미치고 있다. 변동에 따라 애플레이터와 그 주변 통신 소자의 애플레이터를 새롭게 개발하거나 재구성해야 하기 때문이다. 또한 탑재컴퓨터장치 내부구성에 대한 표준화가 미흡하여 변경되는 소자의 모델링과 소자 간의 접속 구현을 위해서 새로운 개발이 반복되고 있다. 게다가 고성능 혹은 다수의 애플레이터의 동시 연산을 위해 시뮬레이터 하드웨어 사양도 증가하고 있다.

• 임무탑재체의 모델의 개발 제약

하나의 우주비행체에 장착되는 임무탑재체의 수와 기능이 증가함에 따라 관련 데이터처리시스템이나 접속장치의 수와 성능도 증가하고 있다. 따라서 시뮬레이터를 구성하는 임무탑재체/접속장치 자체의 기능 구현을 위한 복잡성도 증가하여 모델링 시간이 증가되고 있다. 또한 탑재소프트웨어와 데이터베이스의 규모가 함께 증가하여 최종 버전의 시뮬레이터 개발 비용이 상승하고 있다.

• 시뮬레이터 개발을 위한 입력조건 지체

시뮬레이터를 개발하기 위해 필요한 주요 입력물은 탑재소프트웨어, 데이터베이스, 각종 모델링을 위한 기술자료이다. 이러한 입력물의 지체 혹은 오류는 시뮬레이터 사용자에게 즉각적으로 부정적인 영향을 과급시킬 뿐만 아니라 재검증 비용이 소요된다. 실제 통신

해양기상위성 시뮬레이터 개발 시 탑재소프트웨어의 전달이 2개월 간 지연되어 사용자에게 기존 버전의 시뮬레이터가 전달되었고 필요한 기능의 일부만이 사용되는 제한사항이 존재했다.

한편 우주비행체 개발과정과의 관련성 혹은 종속성과는 별개로 발생할 수 있는 시뮬레이션 소프트웨어의 문제점은 유사한 기능의 수많은 시뮬레이션 모델들을 재차 반복 개발함에 따른 생산성 저하와 유지보수의 어려움에 있다. 다음의 두 항목은 기존에 개발된 소프트웨어 활용의 문제와 미래를 위한 소프트웨어의 표준화 구조의 부재를 설명한다.

• 시뮬레이션 모델 소프트웨어의 표준화 미흡

현재까지 유럽에서 개발된 시뮬레이션 모델은 방대하다. 그러나 기존 소프트웨어를 유럽 내에서 공유하는 것은 소프트웨어 이식 측면이나 사용방법에 있어 난점이 많다. 각기 다른 시뮬레이션 플랫폼을 기반으로 하는 우주기업들 간에 이러한 소프트웨어를 공유하는 방식으로 소스코드(Source Code)를 직접 전달하여 재 컴파일(Compile) 등의 작업을 택하기도 한다. 이러한 일련의 작업은 불필요한 추가 작업비용이 소모되고 공개를 꺼리는 정보를 타 기업에게 제공해야 한다는 단점이 내포한다.

• 시뮬레이션 기반소프트웨어의 표준화 미흡

우주기업들의 개발경험과 산물은 90년대 중반부터 활발하게 일어난 기업 간의 합병에 의해서 시뮬레이터 소프트웨어 개발의 일관성과 효율성을 높이는 계기를 마련하였다. 그러나 'Astrium' 으로 통합된 프랑스와 독일 간의 기존 자원을 이용한 공통 시뮬레이션 기반 소프트웨어가 최근에서야 개발되었으며 유럽 내에서 널리 사용되는 주요 시뮬레이션 기반소프트웨어의 수만 해도 6가지가 넘는 등 여전히 이 부분의 소프트웨어에 대한 표준화는 미흡한 실정이다.

2.3 연구동향 및 전망

앞서 언급된 시뮬레이터 개발과정의 문제점이나 요구사항의 흐름 속에서 최근 10년 동안 유럽에서는

개발 난점을 극복하고 우주비행체 개발에 소요되는 비용을 줄이기 위한 연구를 진행하였다. 특히 유럽 우주국을 주도로 CNES, Astrium, VEGA, SciSys, SPACEBEL, Thales Alenia Space, ELLIDISS, Atos-Origin과 같은 기업이 60만 줄의 방대한 시뮬레이터 소프트웨어의 재사용 연구뿐만 아니라 개발 프로세스, 소프트웨어 설계 요구사항 등의 각종 문서의 재활용에 관한 연구를 어느 때보다도 활발히 진행하였다. 유럽우주국은 시스템 레벨에서 다양한 자원의 재사용을 위해 일관되고 체계적인 기초를 유지하여 시뮬레이션 소프트웨어 개발의 효율성과 실제 적용 및 관리를 통해 실효를 거두고 있다. 최근의 시뮬레이터 연구동향을 살펴보면 다음과 같다.

• 시뮬레이터의 용도범위 확장

시뮬레이터의 용도는 기존의 우주비행체 운용자를 위한 교육 목적 외에 시스템 통합 레벨의 다양한 시험/검증 수단으로 확장되고 있다. 유럽의 “EUROSTAR 3000”을 모체로 하는 통신해양기상위성 개발 단계의 사례에 따라 살펴본 시뮬레이터의 용도 혹은 목적은 다음과 같다.

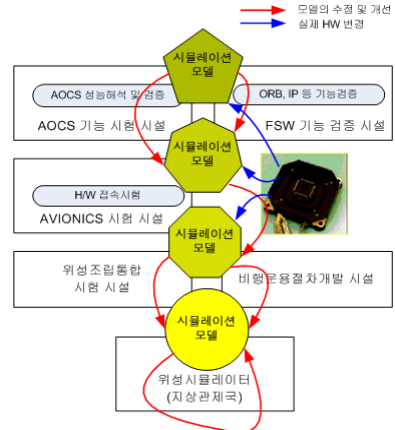
- 탑재소프트웨어 시스템 레벨의 동작검증
- 우주비행체 운용절차서 개발 및 동작검증
- 우주비행체 데이터베이스 일관성 검증
- 우주비행체 조립/통합/시험을 위한 절차서검증
- 지상관제소프트웨어의 기능적 동작검증
- 정상운용/이상상태에 대한 운용자 훈련/교육
- 실제 문제 상황 시 원인분석 및 개선된 탑재소프트웨어 패치(Patch)와 복구운용절차서 검증

사례로 최근 ESTEC의 ‘SIMVIS’ 소프트웨어와 같은 우주비행체 초기개념/임무설계 분석 기능을 시뮬레이터에 포함시키고 있다.

• 단계적인 시뮬레이터 완성화 전략

우주비행체의 개발 중에 시뮬레이터를 점진적으로 개발하여 완성도를 높이는 전략은 90년대 말부터 ESTEC의 범용 프로젝트 시험장치(GPTB)나 갈릴레오 시스템 시뮬레이터에 적용시켜 효용성과 장점을 입증해 왔다. 특히 이러한 전략은 우주비행체를 모사하는 시뮬레이션 모델 소프트웨어를 개발하는

데 유용하다. 초기 우주비행체 설계 단계에서 조립 단계에 이르는 동안 다양한 검증 시설에서 시뮬레이션 모델은 그림 4와 같이 개선되고 진화한다.



자료 : 참고문헌 13 참조
그림 4. 시뮬레이션 모델의 단계적 검증

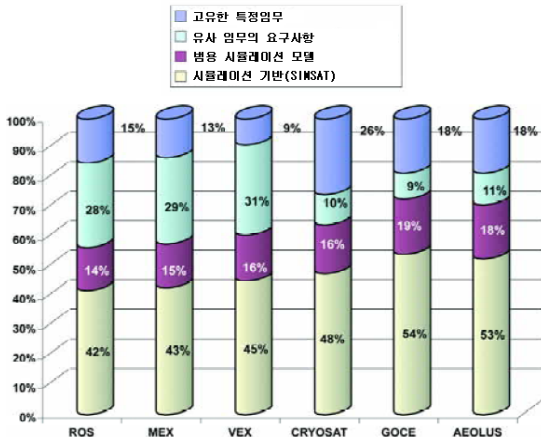
• 입력물 개발일정으로부터 종속성 탈피

최근 시뮬레이터의 입력조건인 탑재소프트웨어/데이터 베이스/기술적인 데이터 등이 생산되기 이전이나 후속 버전이 전달되기 전에 시뮬레이터 개발을 조기에 시작할 수 있도록 간단한 초기 탑재소프트웨어를 생성하여 미리 시뮬레이터에 구성할 수 있는 전략을 선택하였다. ESA/ESOC의 의뢰에 따라 이러한 기능을 갖는 GETS (Generic Emulation Test System)와 같은 도구를 VEGA가 최근에 개발하였다. 이를 이용하면 에뮬레이터, 주변 I/O 모델, 통신버스, 다양한 ASIC 모델을 통합할 수 있어 입력물이 미비한 상태에서도 미리 시뮬레이터 내에 탑재컴퓨터장치 전체의 모델을 구성하여 시간제약이 심한 접속/통합시험을 조기에 완료할 수 있게 되었다. GETS는 최근의 우주비행체인 LISA Pathfinder에 적용된 바 있다. 요구사항의 지체가 발생한 경우의 시뮬레이션 모델도 기능이 유사하고 대체 가능한 기존의 범용 시뮬레이션 모델을 임시 구성하여 입력물 개발일정과의 종속성을 탈피하고 있다.

• 미래를 위한 소프트웨어의 전략적 분류작업

기존의 시뮬레이션 모델의 재사용성, 유지관리를 극대화하기 위해서 소프트웨어 설계 차원과는 별개로 개발 단계별, 우주비행체 계열별, 영역별로 분류 개념을 도입하여 효용 이점을 높이고 있다.

예를 들어, 재사용성을 높이기 위해 ESOC은 다양한 임무를 갖는 우주비행체 시뮬레이터 간의 공통된 요구사항과, 유사한 임무를 갖는 시뮬레이터 간의 공통 요구사항을 도출하였다.



자료 : ESA Bulletin, 2007

그림 5. 우주비행체 임무별 시뮬레이터 소프트웨어 요구사항 분포

그림 5는 행성 간 탐사 임무와 지구 관측 임무 계열로 나누어 임무별 시뮬레이터 소프트웨어의 요구사항 비중을 나타내고 있다.

행성 간 임무 계열로는

- ROS(2004년에 발사한 Rosetta),
- MEX(2003년에 발사한 Mars Express),
- VEX(2005년에 발사한 Venus Express) 와

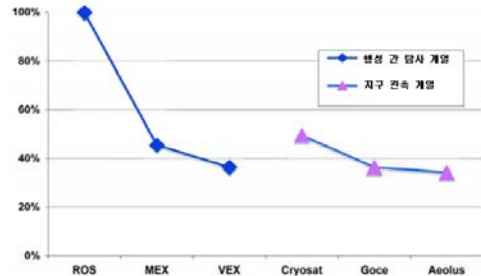
지구 관측 임무 계열로는

- Cryosat(2005년 발사 실패)
- GOCE(2008년 발사 예정)
- Aeolus(2009년 발사 예정)

가 비교되고 있다.

그림 5와 같이 시뮬레이션 기반소프트웨어는 재사용되는 소프트웨어 중 가장 큰 비중을 차지하며 범용 시뮬레이션 모델이 뒤를 이어 50% 이상의 소프트웨어가 완전한 형태로 재사용되고 있음을 보인다. 한편 행성 간 탐사 임무의 유사성에 의해 ROS/MEX/VEX 모두 임무 종속적 소프트웨어가 거의 동일한 비율의 재사용이 효과적으로 이루어진 것을 알 수 있다. 이러한 특징에서 유사 임무의 연속 우주비행체 개발 시 후기에 개발 될 수록 시뮬레이터 개발의 이점이 증가함을 알 수 있음을 알 수 있다. 초기에 개발된 소프트웨어 요구사항은 원형 그대로 다음 프로젝트에 전달되어 어떠한 문서 작업도

할 필요가 없으며 고유한 특정 임무를 위한 소프트웨어 개발에 집중할 수 있다. 이러한 접근방법의 효용성은 그림 6에서 보는 바와 같이 두 번째 시뮬레이터 개발 비용이 급격히 감소하는 것을 통해 입증되었다.



자료 : ESA Bulletin, 2007

그림 6. 분류 효과에 따른 시뮬레이터 개발 비용 변화

그러나 이러한 “임무 계열” 개념에서 연속되는 임무 계열의 프로젝트는 단독 프로젝트보다 수명이 길어서 10년 이상 동안의 소프트웨어 변동성을 고려한 요구사항이 필요하다.

• 임무탑재체 모델의 재현 충실도 증가

운용 지원용이라는 목적에 국한된 시뮬레이터는 태생적으로 임무탑재체의 모델링이 간단하다. 일반적으로 간단한 내부 모드 변화나 TC/TM의 유기적인 처리와 생성만을 모사한다. 그 외의 동작은 모두 시험 전용 시뮬레이터를 추가로 개발하여 사용한다. 그러나 최근 임무탑재체 장치의 기능이 복잡해지면서 운용 지원이나 시스템 시험을 위한 한층 정교한 모델링 요구사항이 전달되고 있다. 예를 들어 최근에 과학실험 목적의 우주비행체인 GOCE나 LISA Pathfinder에 장착된 임무탑재체의 특정 모드에 대한 정교한 동작구현을 요구하고 있다. 이러한 추세에서 복잡한 임무탑재체의 모델의 충실도를 만족시키는 그래픽 모델링 개발도구가 개발되고 있다.

• 시뮬레이션 모델의 표준화

유럽 내에서 기능이 동일한 시뮬레이션 기반소프트웨어 개발 비용을 줄이고 시뮬레이션 모델의 용이한 공유를 위해 기업 간의 협동이 이루어지고 있다. 실질적으로 차세대 시뮬레이터와 시뮬레이션 기반소프트웨어에서 사용될 표준이 정의되고 있는데 2001년에

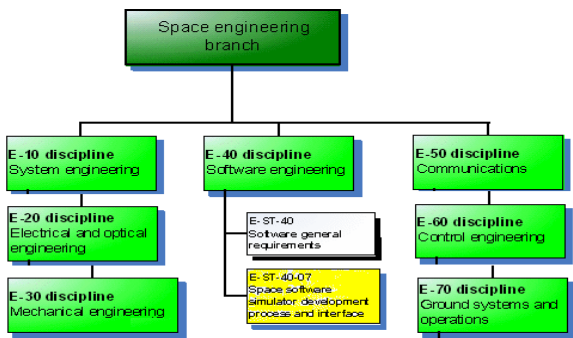
제정된 SMP1(Simulation Model Portability) 표준은 시뮬레이션 모델의 이식성, 유지보수성, 재사용성을 향상시키기 위한 유럽의 실험적인 예비 시뮬레이션 모델링 표준이다. 구체적으로 다음 4가지의 목적을 갖는다.

- 다양한 플랫폼에 대한 시뮬레이션 모델의 상호 작용을 최소화
- 시뮬레이션 모델에서 사용되는 접속의 표준화
- 시뮬레이션 모델 자체의 간단한 접속 실현
- 다른 개발자를 위한 코드 가독성을 증진

SMP를 통해 SMI(Simulation Model Interface) 소프트웨어 템플릿(Template)이 구현되었으며 초기에 시뮬레이션 기반소프트웨어인 EuroSim[3장 참고]과 SIMSAT에 성공적으로 적용되었다.

이후에 SMP1 표준은 시뮬레이션 모델 개발 레벨의 표준이며 SMP1에 시뮬레이션 모델 통합 표준 등을 추가한 SMP2가 제정되었다.

현재 ECSS(European Cooperation for Space Standardization)는 ECSS-E-ST-40-07C(우주용 소프트웨어 시뮬레이터 개발 프로세스와 인터페이스 부문) 부문으로 올해 내에 공식적인 유럽표준 제정을 앞두고 마지막 작업을 진행 중이다.



자료 : <http://www.ecss.nl/>, 2008
 그림 7. 유럽 시뮬레이터 개발 프로세스/접속 표준

ECSS-E-ST-40-07C 표준은 다양한 우주기업들이 개발한 시뮬레이션 기반소프트웨어 사이에 시뮬레이션 모델의 자유로운 이식성을 제공하는 프레임워크(Framework)를 구축하는 데에 목표가 있다. 최근 유럽우주국은 SciSys, SpaceBel, Atos Origin 우주기업에게 다양한 시뮬레이션 기반소프트웨어(SIMSAT 4, BASILES, EuroSim 순서)에 대해 ECSS-E-ST-40-07C의 효과를 증명하기 위한 시뮬레이터 개발을

의뢰하였다.

최근에 ECSS-E-ST-40-07C 표준을 지원하는 시뮬레이션의 모델의 개발과 통합 환경 소프트웨어인 SIMULUS(Integrated Model Development and Integration Environment)가 개발되었다.

그러나 ECSS-E-ST-40-07C 표준에는 시뮬레이터에서 사용되는 시험 용 스크립트나 데이터베이스 파일에 관한 사항이 없어 각 시뮬레이션 기반소프트웨어에서 사용되는 스크립트나 데이터베이스 파일의 공유는 사실 상 불가능하다. 이 부분은 앞으로 해결해야 할 과제이다.

• 분산객체 간의 통신 표준화 이용

플랫폼, 객체위치, 프로토콜에 독립적인 유연한 시뮬레이터를 구성하기 위해 분산객체 간의 통신 표준인 CORBA (Common Object Request Broker Architecture) 지원 미들웨어(Middleware)를 10년 전부터 활발하게 시뮬레이터에 적용하였다. 미들웨어를 통해 다양한 컴퓨터에 분산된 에뮬레이터, 다양한 시뮬레이션 모델, 기반소프트웨어 컴포넌트 간의 데이터 교환을 용이하게 하였고 모델기반의 소프트웨어 설계와 앞서 언급된 ECSS- E-ST-40-07C 표준화에 이점을 제공하였다.

• 리눅스 운영체제로 전환

ESOC는 기존의 윈도우 혹은 솔라리스 기반의 시뮬레이션 소프트웨어를 리눅스 기반으로 전환하고 있다. 리눅스 운영체제를 위한 안정하고 검증된 상용 응용 소프트웨어의 공급이 이루어졌으며 근본적으로 시뮬레이션 환경을 위한 이질적인 하드웨어의 비효율적인 유지보수 문제를 극복하기 위해 플랫폼을 동일화하기 위함이다. 그러나 여전히 플랫폼을 전환하는 과도기 상태이며 위험을 최소화하기 위한 노력을 하고 있다.

• 제 3 소프트웨어와의 접속기능 증가

시뮬레이터 기반소프트웨어의 부족한 기능을 위해 필요한 기능을 유연하게 통합하여 사용할 수 있도록 제 3 소프트웨어와의 접속기능을 확장하고 있다. 우주 비행체 자세제어를 위한 센서/구동기 시뮬레이션 모델은 자세제어 알고리즘을 해석/검증하기 위해 보통 Mathworks 사의 MATLAB이나 MATLAB/Simulink

와 같은 상용 모델링/시뮬레이션/해석 도구를 이용하여 초기에 구현된다. 이러한 시뮬레이션 모델을 그대로 시뮬레이터에 통합시키기 위한 노력이 이루어지고 있다. 이밖에 그래픽스를 위한 PV-WAVE, STK나 모델링 도구인 CAE ROSE, Excel 등의 상용소프트웨어와의 접속을 위한 기능을 추가하고 있다.

• 그 밖의 연구 동향

사용자 요구사항의 잦은 변경에 따른 영향을 최소화 하기 위한 DSDM(Dynamic Systems Development Method) 소프트웨어 기술을 시뮬레이터 개발에 적용을 시도하고 있다.

시뮬레이터의 가속화 성능 혹은 실시간 성능을 만족시키는 데 필요한 하드웨어를 초기에 예측하기는 어렵다. 유럽우주국의 SpaceGRID는 시뮬레이터 요구성능에 부합하는 하드웨어 자원을 예측하고 가상 네트워크 내에 각종 시뮬레이션 자원을 분산시킬 수 있는 도구를 개발하고 있다.

편대 비행 우주비행체를 시뮬레이션 하기 위해 'FTTB'

라는 새로운 시뮬레이션 기반소프트웨어가 설계되고 있다. 여러 대의 시뮬레이터를 구성하고 100에서 200MHz 속도로 동작하는 많은 에뮬레이터를 고성능으로 동작하기 위해 현재 소프트웨어의 구조적인 개념 설계를 진행 중이다.

3. 시뮬레이터 개발 동향

유럽에서 최신의 시뮬레이션 기술이 적용된 시뮬레이션 기반소프트웨어의 특징과 장단점을 논의한다. SIMSAT, SIMWARE, EuroSim 순으로 기술하며 SimTG, Basiles은 간략히 소개한다.

3.1 SIMSAT

SIMSAT은 유럽우주국의 ESOC을 위해 시뮬레이션 분야에 30년 이상의 경험 갖고 있는 VEGA가 개발한 시뮬레이션 기반소프트웨어이다.

표 2. 시뮬레이션 기반소프트웨어 특징 비교

항목	SIMSAT	SIMWARE	EuroSim
시뮬레이션 커널	실시간 기능을 포함한 모든 종류의 시뮬레이터를 위해 개발됨. 다양한 플랫폼을 지원함 : (Alpha/OpenVMS, Window NT, Windows, Linux)	실시간 기능을 포함한 모든 종류의 시뮬레이터를 위해 개발됨. 다양한 플랫폼을 지원함: (VxWorks, Window NT, UNIX, Solaris, Linux)	실시간 기능(IRIX인 경우 최적성능)을 포함한 모든 종류의 시뮬레이터를 위해 개발됨. 다양한 플랫폼을 지원함: (IRIX, Linux, Windows)
시뮬레이션 제어/감시	'Eclipse' 를 사용하여 그래픽 환경 구성 [그림 8 참고], 시뮬레이션 구성용 KAFE 도구	위성조립시험을 위해 개발했던 'OpenCenter' 라는 라이브러리를 기반으로 새롭게 개발됨. 시뮬레이션에 필요한 다양한 기능성 윈도우가 포함됨.[그림 9 참고]	Trolltech 사의 Qt를 이용하여 그래픽 환경 구성. PVWAVE, VEGA, STK 기능 이용.[그림 11 참고]
모델링 도구	상용 CAE ROSE 자체 개발 EGOS-MF, MIE (ECSS-E-ST-40-07C 표준 지원)	'SIMWORK'라는 그래픽 환경의 모델링 소프트웨어가 있음.	그래픽 환경의 모델링 소프트웨어는 없으나 자체 'Model Editor'를 통해 알고리즘 코드. ECSS-E-ST-40-07C 표준 지원
시험용 스크립트	Linux 응용프로그램에서 범용으로 사용되는 Rhino 자바스크립트 이용.	자체 개발한 'LNG'라는 시험 스크립트를 사용.	Linux 응용프로그램에서 범용으로 사용되는 Rhino 자바스크립트 이용.
미들웨어 표준	omniORB의 CORBA 표준	Orbix의 CORBA 표준	사용하지 않으나 일부 연구
에뮬레이터	SIMSAT과 연동하여 검증된 프로세서 에뮬레이터는 1750과 ERC32	SIMWARE와 연동하여 검증된 프로세서 에뮬레이터는 1750과 ERC32, DSP, LEON	SimTG의 ERC32를 연동
기타	MATLAB의 m-file을 Matlab C/C++ 컴파일러를 이용하여 통합가능. 윈도우 용으로 COM기능 사용가능.	ECSS-E-ST-40-07C표준을 지원하지 않음.	MATLAB의 시뮬링 모델 RealTime Workshop이나 MOSAIC이라는 도구를 이용하여 제한된 범위 내에서 사용가능. COM 접속 가능.
개발된 시뮬레이터	ARTEMIS, XMM, Cryosat, Rosetta, Mars Express, Venus Express, Hershel- Planck, Lisa pathfinder, Envisat, Integral, RADARSAT-2 등	INTX, TELESAT, INMARSAT, EUTELSAT, HISPASAT, W3A, AMZ, ANIK F3, ASTRA 1M, SKYNET, HB8, COMS, NIMIQ4 등	European Robotic Arm, NLR Flight simulators, AWETS, ATV, Galileo MUCF AIVP, EuTEF, Gaia RTS 등

SIMSAT은 90년대 중반부터 유럽주우주국의 시물레이터 개발에 사용되었다. 사용자가 시물레이션의 서비스와 모델 실행을 제어하기 위한 KAFE(Kernel Architecture File Editor)라는 도구가 개발되어 빠르게 다양한 시물레이션 환경을 구성할 수 있다는 장점을 지니고 있다. 시물레이션 모델의 개발과 통합을 지원하는 상용 그래픽 모델링 도구인 “CAE ROSE”와 연동이 가능하며 최근 자체 개발한 ‘EGOS- MF’ 모델링 도구는 UML(Unified Modelling Language)로부터 ECSS-E-ST-40-07C에 해당하는 카탈로그를 생성 시키기 위한 도구이며 2장에서 언급된 SIMULUS와 같은 모델 통합 개발 환경(MIE: Modelling Integration Environment)은 시물레이션 모델의 구현/코드생성/스케줄링을 돕는 도구이다. 최근에 개발되어 ECSS-E-ST-40-07C 표준에 맞게 자동으로 C++ 코드를 생성해준다. 단점이라면 상용통신위성과 같은 정지궤도 위성 시물레이터에 적용된 사례가 없다.

최근 SIMSAT은 우주비행체의 초기 개념/임무 설계 시물레이션 기능과 탑재소프트웨어 검증용을 위한 기능을 추가하는 작업을 진행 중이다.

3.2 SIMWARE

프랑스 Astrium에서 10년 전부터 개발한 시물레이션 기반소프트웨어이다. 지난 10년 동안 SIMWARE는 통신 위성 운용을 지원하기 위한 시물레이터로써 세계에서 가장 많이 사용되었다. SIMWARE를 이용하여 현재까지 총 32대의 위성시물레이터가 개발되었으며 2009년에 8대의 개발이 예정되어 있다. 통신해양기상위성의 시물레이터는 SIMWARE를 기반으로 개발되었다. 에플레이터는 프랑스 우주기관인 CNES (Centre National d'Etudes Spatiales)와 프랑스 Astrium이 15년 동안 공동으로 개발하여 왔다. 가장 먼저 개발된 1750 프로세서 에플레이터는 통신위성과 CNES의 PROTEUS의 시물레이터를 위해 개발되어 현재까지도 사용되고 있다.

표 2에서 SIMWARE의 SIMWORK 소프트웨어는 그래픽 환경의 모델링 지원 도구이며 시물레이션 커널이

요구하는 코드를 자동으로 생성해준다. 통신해양기상위성의 복잡한 장치를 신속하고 용이하게 모델링하여 시물레이션 모델을 구현한 바 있다.

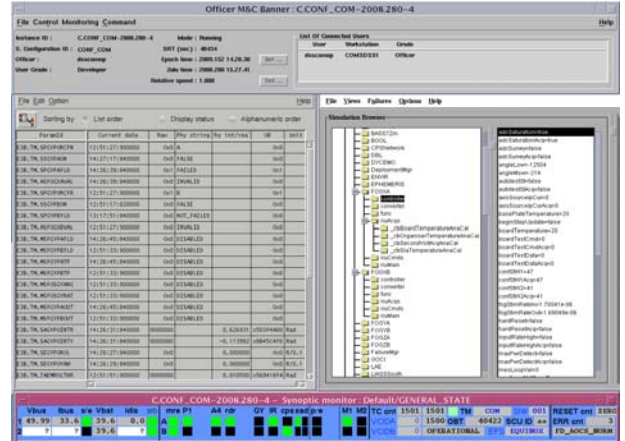


그림 9. SIMWARE: 시물레이션 제어/감시 윈도우

3.3 EuroSim

유럽우주국의 ESTEC에서 빈번하게 사용되고 있는 EuroSim은 초기 우주비행체의 개발과정에서 요구되는 시험/검증을 위한 실시간 시물레이션 시험벤치를 위해 EuroSim 콘소시엄(Atos Origin, EADS Dutch Space, NLR)이 개발하였다. EuroSim은 실제 우주비행체 하드웨어와 연결되는 실시간 상의 하드웨어 패루프(HIL: Hardware in the loop)를 지원하기 위한 목적으로 다양한 외부 하드웨어와의 접속 드라이버가 개발되었다. 모델의 개발/버전제어/추적 등의 미들웨어 장점을 적용하지 않은 이유도 엄밀한 실시간 성능의 구현과 관련되어 있다. 그러나 현재 EuroSim을 구성하는 일부 컴포넌트에 편리하게 통신할 수 있도록 하는 CORBA 적용 연구를 하고 있다. 실제 적용된 사례도 우주비행체 성능시험과 검증을 위한 경우가 대부분의 비중을 차지하지만 ATV (Autonomous Transfer Vehicle) 시물레이터와 같은 지상관제를 지원하기 위한 용도에 성공적으로 적용하기도 하였다.

그림 10은 현재 EuroSim이 단일 프로젝트 내에서 검증시험벤치를 비롯한 모든 종류의 시물레이터로 적용될 수 있음을 보여준다.

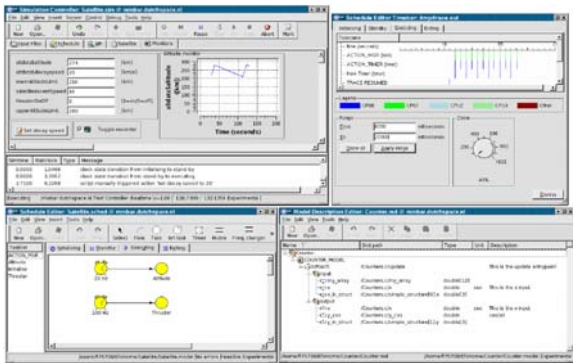
	Pre-Phase A	Phase A	Phase B	Phase C	Phase D	Phase E	Phase F
Feasibility and Performance Analysis/Trade-Offs/Requirements Specification	Conceptual Design Activities	System & Mission Analysis		ATB			
Design Verification		System Interface and End-Design Trade-Offs	System Interface and End-Design Trade-Offs				
Subsystem & Payload View				AV/CSW	AV/CSW		
Spacecraft Qual. and Acceptance				Virtual Test	AV/CSW		
Ground Segment Qual. and Acceptance				MOS Testing	Operations Procedure Validation (OPV)		
System Qual. and Acceptance				AV/CSW	System Maintenance (e.g. S/W)		
Training & Operations				Mission Control Training	On-orbit Training	Investigation of Normal Operations	

자료 : 참고문헌 11

그림 10. EuroSim의 우주비행체 개발 전영역 적용

그림 10의 'CDF' 원은 시스템 엔지니어링 과정에서의 임무/시스템 절충작업과 타당성 분석에서의 적용, 'ATB' 원은 탑재소프트웨어의 검증, 하드웨어 장비의 접속시험/기능시험에서의 적용, 'SOC' 원은 운용 지원을 위한 우주비행체 시뮬레이터로 적용했음을 보여 준다.

EuroSim은 유용한 외부의 기능과 연계하기 위해 MATLAB/Simulink 모델을 MATLAB/Real Time Workshop으로 소스코드를 생성시킨 후 다시 MOSAIC이라는 소프트웨어를 이용하는 방법으로 외부 모델의 이식성을 높이기도 했다.



자료 : <http://www.eurosim.nl/>, 2008

그림 11. EuroSim: 시뮬레이션 개발환경

한편 마이크로소프트 윈도우 플랫폼의 소프트웨어 컴포넌트와 데이터를 교환할 수 있는 COM(Common Object Model)접속을 위한 기능이 내장되어 데이터 감시 및 분석을 위한 클라이언트(Client) 응용프로그램을 이용할 수 있도록 했다. 최근에 EuroSim은 ECSS-E-S T-40-07C 표준의 시뮬레이션 통합 부분을 지원하기 위한 변경 작업을 완료하였으며 사용자 접속 환경을 강화하고 있다.

3.4 기타

독일과 프랑스 Astrium은 기존의 시뮬레이션 기반 소프트웨어를 서로 조화시켜 SimTG를 공동 개발하였다. 시뮬레이션 커널은 프랑스 Astrium의 SIMWARE를 기반으로 시뮬레이션 모델 구조와 초기화 기능 등의 소프트웨어 설계 구조를 향상시켰다. SimTG와 CNES의 Basiles은 SMP2/ECSS E-40-07 표준을 부분적으로 지원하고 있으며 완전한 지원을 위해 추가 작업을 계획하고 있다. 프랑스의 Thales Alenia Space는 K2라는 시뮬레이션 커널을 기반으로 하는 THALASSIM라는 시뮬레이션 기반소프트웨어를 개발하였다. 이것을 기반으로 SPACEBUS 4000 계열의 위성 시뮬레이터를 개발한 바 있다.

4. 결론

유럽의 시뮬레이션 기술은 유럽우주국을 중심으로 하는 시뮬레이션 소프트웨어 재사용 기술의 일관된 연구정책과 방향조정, 생산성 향상을 위한 우주기업들의 다양한 시뮬레이션 도구개발을 통해 한층 더 높은 수준으로 발전하고 있다. 특히, 유럽에서 올해에 제정될 시뮬레이션 모델 표준에 의하여 우주기업들은 생산성 향상에 따른 비용절감 혜택과 국제경쟁력을 갖출 것이다. 이러한 산업 간의 시뮬레이션 모델의 재사용/공유 체계가 성공적으로 이루어짐에 따라 필요한 소프트웨어를 재검파일 없이 단지 바이너리(Binary) 형태의 파일로 원하는 위치에 복사하는 것으로 충분할 때가 올 것이다. 또한 방대한 시뮬레이션 모델 데이터베이스가 구축되어 우주비행체의 고유한 임무를 위한 임무탑재체 모델 개발에 대부분의 시뮬레이터 개발시간이 할당될 것이다.

참고문헌

1. Schumann, H., Maibaum, O., Berres, A., Röhsch, A. "DLR's Virtual Satellite Approach", Netherlands, SESP/ESA, 2008

2. de Vries R., Moelands J., "SMP2 Developments in EuroSim", Netherlands, SESP/ESA, 2008
3. Delatte, B., Manon, F., "BASILES on its Way up to a Wide-spread Simulation Service", Netherlands, SESP/ESA, 2008
4. Eisenmann, H., Cazenave, C., "SimTG: Successful Harmonization of Simulation Infrastructures", Netherlands, SESP/ESA, 2008
5. Sebastiao, N., di Nisio, N., "SMP2 (E-40-07), a New Standard for Simulation Model's Portability and its Implementation in SIMULUS", Netherlands, SESP/ESA, 2008
6. Sarkarati, Spada, Irvine, Ochs, "First Application of the Generic Emulated Test Software, GETS, in the LISA Pathfinder Operational Simulator", Netherlands, SESP/ESA, 2008
7. Guerrucci, Reggestad, Metelo, "Aeolus Floors the Move of ESOC Mission Spacecraft Simulators to Linux", Netherlands, SESP/ESA, 2008
8. Pidgeon, Johns, Atori, Thong Pham, Wormgoor, Lindman, "ECSS E40-07 Demonstration Simulator", Netherlands, SESP/ESA, 2008
9. Laroque, Pecchioli, Sebastiao, "First Experience on Using SMP2 for the Virtual Space System for Ground Data System Validation", Netherlands, SESP/ESA, 2008
10. Nuno S., Vemund R., David., "ESOC Simulators Past, Present and Future", Darmstadt, OPS-G Forum/ESA, 2008
11. Borja G. G., Frank B., Quirien W., "Use of EuroSim in the Modelling and Simulation Section at ESA/ESTEC", Netherlands, 2008
12. Mario M., Alessandro E., Damiano G., Vemund R., David V., "Cutting the Cost of ESA Mission Ground Software", Darmstadt, ESA BULLETIN, 2007
13. 이훈희, 박영웅, 주광혁, 박근주, 양군호, "인공위성 개발과정에서 시뮬레이션 모델의 개발과정 분석", 한국항공우주학회 추계학술대회, 2007
14. Martin J., Verrier D., "The Mars Express Simulator", Darmstadt, VEGA Group PLC, 2004
15. Conrad M., Derek R., "Operational Spacecraft Simulations: Present and Future", Chippenham, IEEE, 2002
16. Soerensen, Rudolph, O'Rourke, Beck, Marc, Pignède, Nergaard, Zanbergen, Suen, Garrido "Flight Operations Segment", ESA BULLETIN, 2001