

무기체계 공통서비스 도출을 위한 IRM과 서비스에디션의 진화과정 분석

옥지흠*, 나윤근*, 윤희병*, 김세일**

*국방대학교 전산정보학과, **국방기술품질원 SW팀

Analysis on the Evolution Process of IRM and Service Editions to Derive Common Service in Weapon Systems

Jiheum Ok, Yungeun Na, Heebyung Yoon, Seil Kim

Korea National Defense University, Defense Agency for Technology and Quality

E-mail : grandojh@hanmail.net, topgun1472@hanmail.net, hbyoon37@hanmail.net, ksismo@hanmail.net

요 약

현대 무기체계의 복잡성과 고가성은 무기체계 개발에 SW컴포넌트인 공통서비스를 도입시켰고, 실례로 미육군에서는 미래전투체계 개발에 공통서비스를 도입하여 개발 중에 있다. 최근 한국군도 무기체계 개발에 공통서비스 도입의 필요성이 제기되고 있다. 따라서 본 논문에서는 무기체계 공통서비스에 대한 선행연구로서 참조모델(RM) 간의 관계 분석과 구현 참조모델(IRM)의 진화에 따른 공통서비스 변화과정을 분석한다. 그리고 서비스 대상의 특성에 따라 3개의 서비스에디션으로 진화된 구현 참조모델의 진화과정과 진화로 인한 영향을 분석한다. 이를 통해 한국군 무기체계 공통서비스를 도출하는데 있어 기술적인 기여를 할 것으로 기대한다.

1. 서론

현대 무기체계는 기능과 역할의 복잡성, 위협성, 핵심기술 유출방지를 위한 비밀성 등과 같은 기술적으로 복잡하고 다양한 특징을 갖는다. 특히 복잡성과 고가성은 무기체계 개발에 있어서 유무인 플랫폼에 장착되는 SW컴포넌트인 공통서비스를 도입하게 되었다. 실례로 미 육군은 미래전투체계의 소프트웨어에 공통서비스를 도입/개발 중에 있다. 최근 한국군도 무기체계 개발에 SW비중이 증가함에 따라 공통서비스 도입의 필요성이 제기되고 있다.

공통서비스는 구현 참조모델(IRM: Implementation Reference Model)을 통해 도출되고, 구현 참조모델은 전략 아키텍처 참조모델(SARM: Strategy Architecture Reference Model)과 서비스 기반 참조모델(SBRM: Service Based Reference Model)을 기반으로 하여

본 연구는 국방기술품질원의 무기체계 내장형 소프트웨어 기술정보 통합관리체계 개념연구 등(과제번호:제 2008_235호)의 연구비지원에 의하여 연구되었음
개발되고 있다. 또한 공통서비스는 구현 참조모델

로부터 서비스 대상의 특성에 따라 3개의 에디션으로 구분되어 진화되고 있다[1].

따라서 본 논문에서는 한국군의 효율적인 무기체계 공통서비스 도출을 위한 미군의 공통서비스 도출 방법인 IRM과 서비스에디션 진화과정을 분석한다.

먼저 2장에서 참조모델간의 관계분석을 하고 3장에서는 구현 참조모델의 진화과정에 따른 공통서비스 변화과정을 분석하며 4장에서는 서비스에디션의 진화과정 분석 및 진화로 인한 영향을 분석한 후, 5장에서 결론을 맺는다.

2. 참조모델 간의 관계분석

공통서비스의 아키텍처 구축을 위해서는 3단계의 참조모델 개발과정을 거친다. 먼저 전략 아키텍처 참조모델을 개발하고 이어서 서비스 기반 참조모델을 개발한 후 이를 이용하여 실제 공통서비스를 구현하는 구현 참조모델을 개발한다.

전략 아키텍처 참조모델에서 서비스 기반 참조모델로 진화되는 과정은 전략 아키텍처 참조모델의 상위계층에 대해서만 집중적으로 구체화하고,

이 과정에서 공통서비스를 정의하고, 무기체계 및 정보체계들에 대한 연동을 정의한다.

서비스 기반 참조모델에서 구현 참조모델로 진화하는 과정은 서비스 기반 참조모델에서 정의한 공통서비스들에 대한 구체화 과정으로 볼 수 있으며, 특히 각 공통서비스들과 이를 구성하는 세부 컴포넌트들에 대한 기능을 정의하고 아키텍처 계층상에서 분포되는 각각의 위치를 정의하고 있다.

참조모델들이 진화됨에 따라서 기술적으로 더 세분화되고 구체화 되었으며, 또한 이러한 참조모델들은 공통적으로 COTS를 기반으로 하고 있고 계층화를 통한 컴포넌트 구조로 분석되고 있다[2].

3. IRM의 진화에 따른 공통서비스 변화과정 분석

공통서비스는 여러 번의 진화과정을 거쳐 서비스 종류가 변경되었다. 구현 참조모델 Ver. 1.0 개발에서는 기존 서비스 기반 참조모델의 서비스 중 워크플로우, 에이전트 및 분석 서비스는 제외되며, 대신 시스템 관리 서비스, 행정 서비스 및 정보 서비스가 새로 정의되었다. 형상 서비스는 형상관리 서비스로, IDM(Improved Data Modem)/통신 서비스는 통신 서비스로 변경되었다. 이러한 서비스 기반 참조모델과 구현 참조모델 간의 서비스 변경사항은 <그림 1>과 같다[1,3].

구현 참조모델 Ver. 1.5는 분석 서비스가 추가되어 15개의 서비스로 정의되었으며, Ver. 2.0은 Ver. 1.5의 지식 서비스에서 제공하던 TIN(Task Integration Network) Manager 및 TDD(Task Description Document) Editor가 태스크 통합 네트

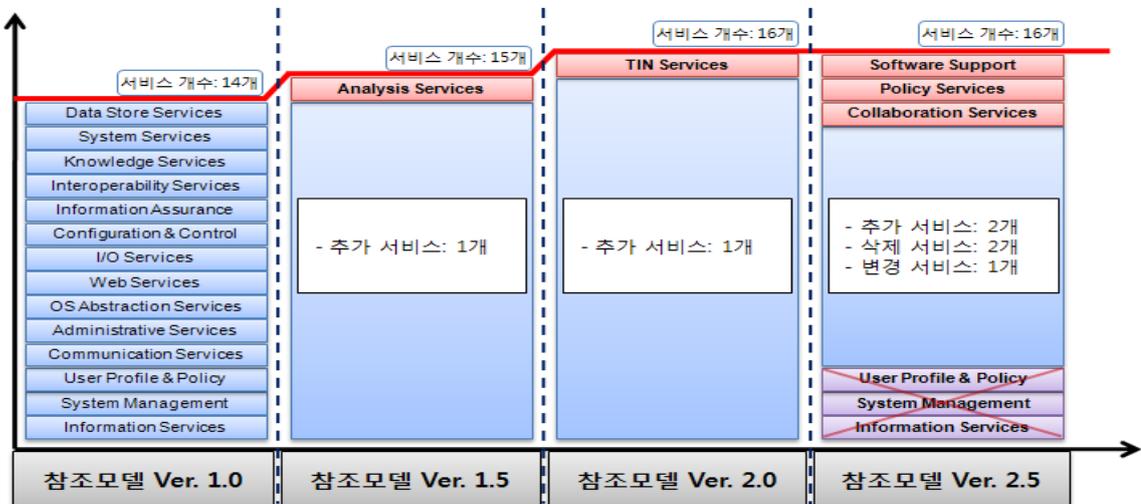
워크 서비스로 분리되어 총 16개의 서비스로 정의되었다.

서비스기반 참조모델 서비스 Family		구현 참조모델 Ver. 1.0 서비스 Family	
Data Store Service		Data Store Service	
System Service		System Service	
Knowledge Service		Knowledge Service	
Interoperability Service		Interoperability Service	
User profile & Policy Service		User profile & PolicyService	
Information Assurance Service		Information Assurance Service	
Configuration Service		Configuration Service	
I/O Service		I/O Service	
IDM/COMM Service		IDM/COMM Service	
Web Service		Web Service	
OS Abstraction Service		OS Abstraction Service	
Workflow Service	삭제	추가	System Management Service
Agent Service			Administrative Service
Analysis Service			Information Service

<그림 1> SBRM과 IRM 서비스 변화

구현 참조모델 Ver. 2.5에서는 사용자 프로파일 및 정책 서비스가 정책 서비스로 통합 변경되었으며, 시스템 관리 서비스와 정보 서비스가 삭제되고 소프트웨어 지원 서비스 및 협업 서비스가 추가되어 지금의 16개의 서비스로 확정되었다. Ver. 3.0에서는 Ver. 2.5와 비교해서 서비스는 동일하나 서비스 구현을 위한 세부 컴포넌트들이 변경되었다 [4,5]. 이와 같은 구현 참조모델의 4가지 버전의 변화 과정을 분석하여 <그림 2>와 같이 제시한다.

구현 참조모델 Ver. 3.0의 진화과정은 2개의 추



<그림 2> IRM 공통 서비스 변화과정

가 서비스, 2개의 삭제 서비스, 1개의 변경 서비스가 있었다. 이를 분석해보면 소프트웨어 지원 서비스와 협업서비스가 추가 되었고, 시스템 관리 서비스와 정보 서비스가 삭제되었다. 그리고 기존에 사용자 프로파일 및 정책서비스로 한 개의 서비스로 제공하였으나 진화과정에서 사용자 프로파일은 제외되어 정책서비스만 남게 된 것으로 분석되었다.

구현 참조모델 버전 변경에 따라 아키텍처 계층도 변경되었는데 이를 분석해보면 구현 참조모델 Ver. 1.5까지는 세부 서비스(14개)가 개념적 정의로 각각 하나의 계층에 포함되어 있었으나, Ver. 2.0부터 공통서비스는 이를 구성하는 다수 컴포넌트들의 묶음으로 구현되었고, 컴포넌트들이 여러 계층에 분산되어 분포되었으며 통합 서비스계층이 새로 추가되었다.

Ver. 2.0과 Ver. 3.0에서 공통 도메인 서비스 계층이 공유 도메인 서비스 계층과 혼합 기반구조 서비스 계층으로 분화되었다. 각 아키텍처 계층별 서비스 내용 및 변경 현황은 <표 1>과 같다.

<표 1> IRM 버전별 아키텍처 계층 비교

Ver. 1.0	Ver. 2.0	Ver. 3.0
- 도메인 애플리케이션 계층(Domain Application Layer) • 전장명령, 상황인식에 속하는 도메인 특정 컴포넌트 포함		
-공통도메인 서비스 계층(Common Domain Service Layer) • 컴포넌트에 의해 사용되는 일반화된 글로벌 도메인 서비스 제공	- 공유 도메인 서비스 계층 (Shared Domain Service Layer) • 다른 도메인이 요구하는 일반화된 서비스 제공	
	- 혼합 기반구조 서비스 계층 (Composite Infrastructures Service Layer) • 시스템 편의성 및 기능성을 제공하기 위한 컴포넌트의 집합 제공	
없음	-통합 서비스계층(Integration Service Layer) • 사용자와 어플리케이션 뒷단에서 동작하는 상황감지 서비스 제공	

4. 서비스에디션 진화과정 분석

4.1 서비스에디션 진화과정

구현 참조모델은 서비스 대상에 대한 특성을 고려하며 이를 지원하는 공통서비스들이 그룹핑되어 다양한 에디션을 이루게 된다. 이런 다양한 에디션은 성능, 확장성, 이식성, 조립성 및 상호운용

성의 요구사항을 충족시킨다. 구현 참조모델에 있어서는 표준 에디션, 실시간 에디션, 마이크로 에디션인 3가지 에디션으로 구분되며 그 중 표준 에디션은 가장 강건한 전투 지휘 형상을 지원한다. 그리고 실시간 에디션은 결정적인 단일 프로세스 및 다중 쓰레드 환경에서 요구되는 서비스를 제공하며, 마이크로 에디션은 무인지상센서(UGS)와 같은 소형의 신호와 데이터 프로세서를 지원한다.

구현 참조모델의 진화과정을 중심으로 하여 그 진화과정 내에서 서비스 에디션의 진화과정을 분석해보면, 공통서비스에 대한 에디션별 설계는 여러 사항을 포함하고 있고 프로그램 코드는 에디션 간 재사용이 가능하다. 또한 서비스 제공을 위한 API는 표준화되어 있어 각각 다른 에디션에 대해 상호운용이 가능하다.

Ver. 2.5의 경우 표준, 분해(Dismount), 실시간, 마이크로의 4가지 에디션으로 구분되었으나 Ver. 3.0의 경우 분해가 제외된 3가지 에디션으로 구분되는 차이점이 있다. Ver. 2.5와 3.0 간의 에디션별 역할 및 세부 변경현황을 <표 2>와 같이 제시한다.

<표 2> 에디션별 IRM 변경사항 분석

서비스군	실시간		마이크로	
	Ver. 2.5	Ver. 3.0	Ver. 2.5	Ver. 3.0
소프트웨어 지원 서비스		○		○
TIN 서비스	○			
I/O 서비스	○			
데이터 저장 서비스	○	○	○	
정책 서비스	○	○	○	
※ 3개 서비스 에디션 모두 운영체제 추상화 서비스, 시스템 서비스, 형상 및 제어 서비스, 정보 보증 서비스 및 통신 서비스, 상호운용성 서비스(실시간)를 포함하고 있음				
※ 표준 에디션은 변동 없음				

표준 에디션은 분석 서비스의 데이터 마이닝 컴포넌트가 데이터 마이닝과 텍스트 마이닝 컴포넌트로 분리되고, 도메인 애플리케이션 계층에 위치하던 TIN서비스의 TDD Editor 컴포넌트가 통합 서비스 계층으로 이동되었다.

실시간 에디션은 Ver 3.0에서 TIN서비스 및 입/

출력 서비스는 삭제되고 소프트웨어 지원 서비스가 새로 추가되었다.

마이크로 에디션은 Ver. 2.5 대비 3.0에서는 컴포넌트 수가 감소하였고, Ver. 2.5의 경우 마이크로 에디션이 표준 에디션과 동일한 서비스를 제공해야 하는 경우 서비스를 구성하는 컴포넌트가 거의 동일하였으나 Ver. 3.0에서는 동일한 서비스를 제공하더라도 필요 기능의 컴포넌트 외에는 제외되었다.

4.2 서비스 에디션 진화로 인한 영향 분석

구현 참조모델이 Ver. 2.5에서 Ver. 3.0으로 진화하는 과정에서 에디션별로 공통서비스의 증감 또는 세부 컴포넌트들의 변동이 있었다. 따라서 그 변화를 통해 각 에디션이 지원하는 애플리케이션들의 능력변화에 대해 분석한다.

표준 에디션은 공통서비스의 증감과 같은 큰 변화는 없다. 이를 통해 표준 에디션의 역할이 초기에 정립된 대로 유지되어 진화과정에서도 능력의 변화가 없음을 알 수 있다.

실시간 에디션은 소프트웨어 지원 서비스의 추가를 통해 공통정보를 관리하며 모드관리, 시스템 초기화, 자원할당을 할 수 있도록 한다. 이는 급변하는 전장상황 하에 있는 무기체계가 실시간으로 기민하게 대처할 수 있는 기능을 추가한 것이다. 삭제된 서비스로 태스크 통합을 통해 시분할로 처리하는 네트워크 서비스가 있는데 이는 실시간을 지원하는데 있어서 불필요한 것으로 분석되었고 또한 애플리케이션의 인터페이스가 표준화되면서 정보 교환을 입/출력 처리를 통해 수행하는 것이 불필요하게 되어 제외된 것으로 분석된다.

마이크로 에디션은 실시간 지원이 반드시 필요한 요소이므로 소프트웨어 지원 서비스의 추가는 실시간 에디션과 같을 것으로 분석된다. 마이크로 에디션이 지원하는 센서, 로봇은 최전방에서 적과 접촉한 상태에서 정보를 수집 및 전투를 수행하는데 있어서 판단, 결심보다는 첩보수집과 전송 그리고 지시를 받는 구조로 단순화가 필요하게 되었다. 따라서 데이터 저장 서비스는 불필요한 요소로 판단되어 삭제한 것으로 분석된다. 또한 각 무기체계별로 "Plug and Play" 개념을 적용하고 있어 애플리케이션 중심에서 그 특성에 적절한 서비스들을

제공받는 시스템이 요구되므로 사용자 프로파일을 관리하는 정책서비스는 불필요한 것으로 분석된다.

5. 결론

본 논문은 무기체계 공통서비스 도출을 위한 구현 참조모델과 서비스 에디션의 진화과정을 분석하였다. 관계분석을 통해 공통서비스의 기반이 되는 구현 참조모델이 전략 아키텍처 참조모델과 서비스 기반 참조 모델을 개발한 후 이를 이용하여 개발됨을 알 수 있었다.

구현 참조모델의 진화는 필요에 따라 공통서비스의 증감 또는 변경이 있었으며 이 진화는 서비스 에디션들의 진화에 영향을 끼쳤다. 이러한 진화과정을 통해 국방 분야 연구개발에 있어서 급변하는 기술 환경과 요구사항은 진화과정이 필연적임을 알 수 있었다. 따라서 본 논문에서 분석한 진화과정이 향후 무기체계 공통서비스를 도출하는데 있어서 기술적인 기여를 할 것으로 기대한다.

[참고문헌]

- [1] Paul Schoen, "System of Systems Common Operating Environment(SOSCOE)" NCO Advanced Software, IDeAS, PW, December 6, 2004.
- [2] Paul Scheem, "System Of Systems Common Operating Environment,(SOSCOE) 'Use Of COTS/Open Source Software'", FCS Boeing, September. 2006.
- [3] Mike Williamson, Matt Schnaidt, "Future Combat System(FCS) System of Systems Common Operating Enviroment (SOSCOE)," AFCEA Conference, November 9, 2006.
- [4] Mike Williamson, Matt Schnaidt, "Future Combat System(FCS) System of Systems Common Operating Environment (SOSCOE)," AFCEA Conference, November 9, 2006.
- [5] Mike Williamson, "Future Combat Systems / SOSCOE", Network Systems Integration, FCS, November 9, 2006.