

## A Study on the Standard Architecture of IFF Interface SW in the Naval Combat Management System

Yeon-Hee Noh\*, Dong-Han Jung\*, Young-San Kim\*, Hyo-Jo Lee\*

\*Engineer, Naval R&D Center, Hanwha Systems, Pangyo, Korea

### [Abstract]

In this paper, we propose the standard architecture for the IFF interface SW in naval combat management system(CMS). The proposed standard interface architecture is a method designed to reduce modification efforts and man-month of reliability test for the existing the IFF interface SW of 11 types. We identified highly dependent CMS and GFE information, leading to the redefinition of standard requirements and functions, and proceeded with the initial design applying the Naval Shield Component Platform(NSCP). Subsequently, using the Feature Model, we derived additional common and variable elements for the interface of multiple CMS and GFE. Considering the S.O.L.I.D principles, we designed the final architecture. The proposed IFF Interface SW, based on the standard architecture, is expected to enhance management efficiency through a common architecture, increase code reusability and scalability, and reduce development costs by shortening reliability testing times.

▶ **Key words:** Naval Combat Management System, IFF Interface SW, Standard Interface Architecture, Feature Model, SW Reuse

### [요 약]

본 논문은 함정 전투체계 피아식별기 연동 소프트웨어에 대한 표준 연동 아키텍처를 제안한다. 제안된 표준 연동 아키텍처는 기존의 11종 피아식별기 연동 소프트웨어의 수정 공수 및 신뢰성시험 수행시간 단축을 위해 고안된 방법으로, 의존도가 높은 체계 및 장비 정보들을 식별하고 표준 요구사항 및 기능을 새로 정의하였다. 이를 기반으로 차기 함정전투체계 표준 연동 아키텍처(NSCP)를 적용해 1차 설계를 진행하였다. 이후 휘처 모델(Feature Model)을 통해 다중 체계 및 장비 연동에 대한 추가 공통요소와 가변요소를 도출하고 S.O.L.I.D 원칙을 고려해 최종 아키텍처를 설계하였다. 제안된 피아식별기 표준 연동 SW는 공통 아키텍처를 통한 관리 효율성 증대와 소스코드의 재사용성 및 확장성 증가, 신뢰성시험 수행시간 단축을 통한 개발비용 감소 효과를 기대할 수 있다.

▶ **주제어:** 함정전투체계, 피아식별기 연동 SW, 표준 연동 아키텍처, 휘처 모델, SW 재사용

- First Author: Yeon-Hee Noh, Corresponding Author: Yeon-Hee Noh
- \*Yeon-Hee Noh (yhee.noh@hanwha.com), Naval R&D Center, Hanwha Systems
- \*Dong-Han Jung (konho42@naver.com), Naval R&D Center, Hanwha Systems
- \*Young-San Kim (y.san.kim@hanwha.com), Naval R&D Center, Hanwha Systems
- \*Hyo-Jo Lee (hjwish.lee@hanwha.com), Naval R&D Center, Hanwha Systems
- Received: 2023. 12. 08, Revised: 2024. 01. 02, Accepted: 2024. 01. 15.

## I. Introduction

함정 전투체계(CMS, Naval Combat Management System)의 소프트웨어 중 연동 소프트웨어는 센서 및 무장 등의 장비를 체계에 연동시키는 역할로, 각기 다른 함정 장비를 체계에서 원격으로 운용 및 통제하는 기능을 지원하는 소프트웨어이다. 따라서 함정 전투체계의 구성과 대상 장비의 연동 정보의 변화에 따라 SW 수정이 필연적으로 수반되는 특성을 가진다. 이러한 의존성은 개발 비용의 증가를 야기한다.

현재 개발 중인 많은 무기체계에서 SW 기반 무기체계의 설계와 개발이 이루어지고 있으며, 피아식별기 연동 소프트웨어와 관련된 IFF Mode-5 성능개량 사업 또한 그중 하나이다. 해당 사업의 목적은 HW의 변경은 최소화하고 SW를 변경하여 아군의 피아식별 능력을 향상시키는 것이다. 이러한 성능개량 사업은 변경에 의한 영향성을 최소화 하면서 기존 장비의 작전운용성능과 소요군의 요구사항을 충족시키는 고품질의 SW 개발을 목표로 한다[1]. 이를 위해서는 피아식별기 연동 소프트웨어의 품질 개선이 필수적이지만 현재의 피아식별기 연동 소프트웨어는 대상 함정과 연동 장비의 범위가 커질수록 개발비용도 비례하여 증가한다는 문제점이 있다. 특히, 소스코드의 변경은 소프트웨어의 재설계 및 구현, 기능시험(단위시험, 개발시험, 운용시험)과 더불어 소프트웨어 정적 시험(MISRA[2], CWE, Metric) 및 동적 시험(Statement Coverage)이라는 SW 신뢰성 시험 수행을 동반하며[3], 이는 많은 노동력과 시간을 요구한다. 또한, SW 수정 및 신뢰성 시험의 수행은 소스코드 규모 및 난이도, 개발자의 숙련도, 보조 도구 사용여부, 신뢰성시험 검증 툴의 라이선스 수 등 업무 수행에 영향을 미치는 요소들을 가지고 있으며 사람에 의한 작업으로 인적 오류의 잠재적인 요소도 존재한다[4].

이러한 문제를 해결하기 위해서는 각기 다른 함정의 피아식별기 연동 소프트웨어들을 위한 단일화된 형상이면서 함정 및 장비의 가변성에도 재사용성을 높일 수 있는 표준 연동 아키텍처가 필요하다. 이는 궁극적으로는 고품질과 저비용이 동시에 만족하는 SW 설계 개발을 의미한다. 본 논문에서는 시제 함정 11종에 탑재된 피아식별기 연동 소프트웨어들을 보다 효율적으로 대체하기 위한 피아식별기 연동 SW 표준 아키텍처 설계 연구를 제시하였다. 차기 함정전투체계를 위해 제안된 표준 연동 아키텍처 연구를 바탕으로 기존의 구형 피아식별기 연동 SW들을 단일화된 표준 형식으로 신규 설계하였다. 또한 설계된 표준안을 대상으로 공통요소와 가변요소들을 파악하는 휘처 모델을

추가 적용하여 피아식별기 표준 연동 구조를 개선하였으며, 기존 소프트웨어와 비교한 결과를 기술하였다.

본 논문의 구성은 다음과 같다. 2장에서는 현재 함정 전투체계 소프트웨어의 구조와 표준 연동 아키텍처 및 관련 연구에 대해 소개한다. 3장에서는 새로운 피아식별기 연동 SW 표준 아키텍처 설계에 대해 설명한다. 4장에서는 본 논문에서 새롭게 설계한 피아식별기 연동 SW와 기존 SW의 개발 요소를 비교하고 평가한다. 마지막으로 5장에서는 연구결과에 대한 결론으로 논문을 마무리한다.

## II. Preliminaries

### 1. Related works

#### 1.1 Naval Combat Management System

함정 전투체계는 Fig. 1과 같이 함정에 탑재된 모든 센서, 무장 등 외부로부터 획득된 정보를 종합 처리하여 최적의 전투능력을 제공할 수 있도록 지휘 및 무장통제를 자동화하는 통합시스템이다. 국산화 및 국산 함정전투체계의 배치 이후 지속적인 방산 무기체계 발전과 함께 전투체계 소프트웨어의 변경 또한 수시로 발생되어 왔다[5-6].

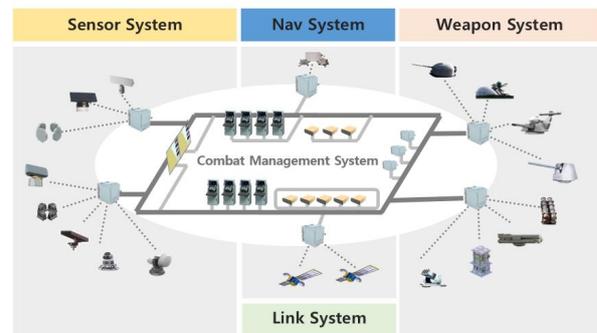


Fig. 1. System Architecture of Naval Combat System

#### 1.2 Interface Control Unit and Interface SW

연동단(ICU, Interface Control Unit)은 센서 및 무장을 연동시키는 역할을 하며, 일반적으로 관급장비(GFE, Government Furnished Equipment)와 CMS 간 상호 통신을 담당한다. GFE 통신은 통신 프로토콜, 주기, 데이터 형식 등의 합의된 연동 사양을 규정한 ICD에 따라서 다른 두 체계 간 데이터 및 명령을 교환하는 기능을 수행하는 것을 의미한다[7]. ICU와 일대일로 연결되는 것이 보편적이며, 기존에 양산된 장비의 경우 해당 정보가 이미 확정되어 변경이 어렵다. 반면에 CMS 통신은 Fig.3 과 같은 메시지 기반의 DDS(Data Distribution Service)[8] 통

신을 사용하여 데이터를 주고받는 다중 통신 구조이다. ICU는 전투체계 데이터 버스(CSDB, Combat System Data Bus)에 연결되어 필요한 CMS 메시지를 송/수신할 수 있으나, 전투체계에 탑재된 DDS의 미들웨어 형상에 따라 그 방식이 상이할 수 있다. DDS 특성상 해당 통신 관련 클래스들 기본적으로 포함해야 하며, 메시지 변경 시 관련 클래스 또한 반드시 변경되어야 한다[9].

이러한 연동단 역할을 수행하는 CMS 소프트웨어가 바로 연동 SW이므로, 단순히 탑재되는 연동단의 OS 뿐만 아니라 CMS 환경과 GFE 환경 모두에 의존적일 수 밖에 없는 형상이다.

### 1.3 IFF Mode-5 Product Improvement

주로 질문기와 응답기로 이루어지는 피아식별기(IFF, Identification Friend or Foe)는 전파 또는 음파를 이용해 적과 아군을 식별하는 암호장비이다. 공중 및 수상 이동수단 등 다양한 무기체계에서 사용된다. 이러한 피아식별기의 Mode-5 성능개량이란, 한미 연합 작전수행 능력 향상을 위해 피아식별기의 식별 모드를 비화기능이 강화된 Mode-5로 전환하는 과제를 의미한다. 과거 북대서양 조약기구(NATO)를 비롯한 친 서방권 국가들은 Mode 1, 2, 3/A, C, 4 방식의 피아식별모드를 사용해왔으나 구형 암호 알고리즘이 적용된 Mode-4의 보안 취약점이 발견되면서 최근에는 Mode-4를 중단하고 신형 Mode-5를 적용하기로 결정되었다. 이에 따라 한국 군 역시 Mode-5로의 전환을 진행중이다[10].

### 1.4 NSCP(Naval Shield Component Platform)

해군 함정전투체계의 두뇌라고 할 수 있는 CMS 소프트웨어는 안정성이 중요시되는 방위산업의 특성 상 초창기 설계된 절차지향형 구조를 유지해왔다. 특히 연동 소프트웨어는 연동 무기체계에 의존적인 형상이므로 전체 CMS 소프트웨어에 대한 체계통합 구현에 많은 제한이 있었다.

NSCP(Naval Shield Component Platform)는 이러한 문제를 해결하기 위해 제시된 차기 함정전투체계 표준 연동 소프트웨어 아키텍처이다. 연동 소프트웨어의 공통적인 주요 기능을 4개의 컴포넌트로 분류하고 S.O.L.I.D 설계 원칙과 디자인 패턴을 일부 적용하여 단일 함정 전투체계에 대한 표준 아키텍처를 제안하였다[11]. 인접한 두 컴포넌트는 송/수신 별 독립적인 인터페이스를 사용하여 상호 유관 정보를 전달한다. 전체 컴포넌트의 주요 기능을 요약하면 아래와 Table 1과 같다.

Table 1. Naval Shield Component Platform

Component Name	Major Function
Interface Manager	External Communication - CMS / GFE Interface
Message Converter	Data Converting - Msg <-> Info
Function Manager	Data Classification & Distribution
Subfunction Manager	Data Processing

저자 C.S. Baek, J. H. Ahn의 연구[11]에서 NSCP는 기존 대비 공용 클래스 포함 비율의 58% 증가, 단일 메시지 변경 시 구조변경 비율의 25% 감소됨을 확인하였고, 높은 유지보수성과 재사용성 향상 효과를 기대할 수 있었다.

다만 다수의 함정 및 장비를 대상으로 하는 과제에서 해당 아키텍처를 사용할 경우, 현재 정의된 공용 컴포넌트에서 추가적인 가변성이 발생할 여지가 있어 바로 적용하기는 어렵다. 따라서 본 논문에서는 해당 아키텍처를 기반으로 하여 표준 설계 구조에 대한 이점은 보유하되, 실제 연동 대상 특성에 따라 발생가능한 변경요소의 범위를 추가적으로 판단하여 보다 개선된 설계를 적용하고자 하였다.

### 1.5 Feature Model and FORM

소프트웨어의 공통성과 가변성을 분석하기 위해 사용되는 도메인 분석 방법 중 하나인 FORM(Feature-Oriented Reuse Method)은 FODA에서 제안한 휘처 모델로서, 특정 도메인의 가변성을 식별하고 그것을 반영한 재사용 컴포넌트를 만들어 개별 제품을 빠르게 생성해내는 장점이 있다[12]. 휘처 모델을 바탕으로 AND/OR 그래프를 이용해 공통성을 추출하며, 이때 AND 노드는 제품의 필수적 휘처, OR 노드는 타 제품으로 선택될 수 있는 양자택일의 휘처를 뜻한다[13].

본 논문에서는 IFF M.5 대상 시제인 기존 함정 11종 피아식별기 연동 SW의 도메인에 대해 휘처 모델링하여 공통요소와 가변요소를 효과적으로 도출하였고, NSCP 기반의 표준 연동 아키텍처로 선언된 공용 클래스에서 추가적인 가변요소를 찾아 분리되도록 설계하였다.

## III. The Proposed Scheme

### 1. Standard Architecture of IFF Interface SW in the Combat Management System

피아식별기 연동 SW 표준 아키텍처로 재설계하는 과정은 아래와 같이 4단계로 진행하였다.

- Step 1. Identification : 기존 피아식별기 연동 SW의 대상 함정과 장비를 성격에 따라 분류하고, 피아식별기 연동 SW의 표준화된 요구사항과 기능을 식별하였다.
- Step 2. First Design : 식별된 기능을 기준으로 NSCP 기반의 표준 연동 아키텍처 설계를 적용하였고 해당 결과를 분석하였다.
- Step 3. Classification : 휘처 모델을 통해 1차 설계된 공통요소 항목 중 추가적인 가변요소 여부를 판단하였다.
- Step 4. Final Design : 추가적으로 발생한 가변요소에 대해 개선된 설계를 적용하였다.

**Step 1. Identification of existing IFF interface SW**

**Step 1.1. Identify CMS type of existing IFF interface SW with CMS baselines and middlewares**

피아식별기 연동 SW 표준 아키텍처 구현 시 고려되어야 할 주요 항목을 기준으로 대상 함정을 재분류하였다. 여기서 대상 함정은 IFF Mode-5 성능개량 대상의 해군 함정 전력 11종을 의미하며, 함정 종류 별 대상 전력은 다음과 같다.

- 구축함 2종(KDX-I PIP, KDX-II)
- 호위함 2종(FFX Batch-I, FFX Batch-II)
- 고속함 1종(PKX-A) 및 고속정 1종(PKX-B)
- 대형수송함 2종(LPX #1, LPX #2)
- 상륙함 1종(LST-II)
- 기뢰부설함 1종(MLS-II)
- 훈련함 1종(ATX)

다양한 방식의 분류법이 있겠지만 본 논문에서는 대상 소프트웨어의 표준화 구현 시 가장 큰 영향을 미치는 요소들만 고려하였다. 먼저 CMS Baseline을 기준으로 대상 함정을 크게 네 가지 유형으로 구분할 수 있다. 여기서 Baseline은 Fig. 2와 같이 국내 전투체계 발전의 흐름을 의미하며[14], 그에 따라 체계의 모듈 설계 구조, 미들웨어, 연동 장비 등이 함께 업그레이드되어왔다.



Fig. 2. Baselines of Naval Combat Management System

그중 체계의 구조는 연동 SW의 표준 설계 시 큰 영향을 주는 요소 중 하나이다. 특히 Baseline 1.0의 경우, 센서 또는 무장을 위한 상위 통합모듈 SW와 개별 연동 SW가 함께 존재하면서 장비 연동 처리 기능이 분산되어 있었다. 이러한 설계구조는 Baseline 2.0으로 진화하면서 개별 연동 SW로 단일화되었고, 전보다 재사용성과 확장성이 향상되었다. 따라서 본 논문에서도 기존 CMS 형상을 위배하지 않는 선에서 Baseline 1.0의 피아식별기 연동 SW 기능을 최신 구조에 맞춰 이관하는 단일화 작업을 진행하였다.

다음으로, CMS는 Baseline에 따라 각기 다른 체계 미들웨어를 사용해왔다. 하지만 표준화된 체계 통신 구축을 위해서는 이를 공용으로 처리할 수 있는 표준형 체계 통신 어댑터 기능이 필요하다. 특히 외산함의 경우 외산 CMS가 탑재되어 있고 그에 따른 전용 미들웨어를 사용하므로 국내 전투체계 시스템과의 호환성은 지원되지 않는다. 따라서, 외산 미들웨어와 국내 미들웨어 사이에서 브릿지 역할을 하는 별도의 모듈을 추가하여 피아식별기 연동 SW의 표준성을 보장하는 작업이 필요하였다. 이러한 특성적 한계로, 본 논문에서는 내부 확인이 불가능한 외산함 CMS는 비교 대상에서 제외하였다. 하지만 향후 외산함의 CMS 단위 성능개량 사업이 추진된다면 기존 브릿지 모듈만 제거하고 제안하는 피아식별기 표준 연동 SW를 그대로 활용할 수 있다.

또한 Baseline 2.0 내부에서도 미들웨어의 변경으로 인한 체계 통신기능 설계의 차이가 존재하였다. 기존의 미들웨어 RTI DDS는 국제 표준 기술인 DDS[8]를 기반으로 효율적인 다중 통신 구조를 제공하는 상용 미들웨어이다. 이후 개발된 Smart DDS[15]는 함정전투체계라는 도메인 특성을 고려하여 한 단계 더 발전한 미들웨어라고 할 수 있다. 함정에 맞춰 보다 경량화된 데이터 통신과 체계 구성 변화에 대한 높은 확장성을 지원한다. 따라서 피아식별기 표준 연동 SW 구현 시에는 이점을 유의하여 설계하였다. 반면, 장비 통신을 위한 미들웨어(PSC)는 별도의 변경 없이 단일 형상으로 사용 가능하였다. 이러한 체계 Baseline 별 미들웨어 및 함정에 관한 설명 요약은 Table 2와 같다.

Table 2. CMS Type Description with Baselines and Middlewares

Baselines	CMS middleware	CMS Name
foreign CMS	foreign middleware	KDX-II, LPX #1
Baseline 1.0	RTI DDS X.X	PKX-A
Baseline 2.0	RTI DDS X.X	FFX Batch-I, LST-II, MLS-II
	Smart DDS X.X	FFX Batch-II, PKX-B, ATX, LPX-#2, KDX-I PIP

**Step 1.2. Identify GFE type of existing IFF interface SW with CMS name**

다음으로는 피아식별기 연동 SW의 연동 대상이 되는 피아식별기 장비 자체의 유형을 분류할 수 있다. IFF Mode-5 성능개량 과제에서는 Table 3과 같이 피아식별기 장비 업체에 따라 3개로 유형을 구분하여 추진하였다.

Table 3. GFE Type Description with CMS name

GFE Type	CMS Name
IFF Type-I (domestic, A)	KDX-II, LPX #1, KDX-I PIP, PKX-A
IFF Type-II (foreign, B)	MLS-II, FFX Batch-II, ATX, LPX-#2
IFF Type-III (foreign, C)	FFX Batch-I, LST-II, PKX-B

대부분의 함정에 탑재된 IFF 장비는 일부 HW 또는 SW 업그레이드를 통해 Mode-5 지원이 가능하도록 성능 개량이 가능하였다. 그러나 일부 유지보수가 불가능 구형의 외산 장비인 경우 이를 대체하기 위한 신규 IFF 장비 탑재가 불가피했고, 이를 위해 국내의 장비 업체 A사에서 신규 개발된 유형이 바로 IFF Type-I형이다. Type-I형은 기존에 국내에서 많은 비중을 차지하던 해외의 장비 업체 B사의 IFF Type-II형과 유사성이 높게 설계되어 전체 함정 내 IFF 장비의 표준성이 향상되었다.

반면, IFF Type-III형(해외, C사)의 경우 질문기/응답기 기능이 일체형으로 제공되므로 대상 업체가 버전을 관리한다는 점에서 I, II형과는 다르다. IFF Type-I(국내, A사)형과 IFF Type-II형(해외, B사)은 질문기/응답기가 모듈화되어 각각 대상 업체와 버전이 별도로 존재한다.

또 한편으로는 피아식별기의 고유 특성 면에서 I, II, III형은 필수적으로 공통적인 요소도 존재한다. 따라서 단순히 장비 유형의 구분만이 아니라 하위 기능적 특성을 세세히 파악하여 표준화/공용화를 진행하는 것이 바람직하다고 판단하였다.

**Step 1.3. Identify function of existing IFF Interface SW**

피아식별기 연동 SW는 피아식별기와의 네트워크 연결을 토대로 필요한 연동 메시지를 송수신해야 한다. 일반적으로 피아식별기로 자함 정보, 연동 요청, 질문기/응답기 통제 등 체계의 정보 혹은 명령을 전달하는 기능을 수행하고, 피아식별기로부터 장비의 상태정보와 표적정보 등의 수신정보를 처리하여 체계로 전달하는 기능을 수행한다. 이때 기존 함정에 탑재된 피아식별기 성능에 따라 요구사

항 및 기능의 보유 여부는 각기 상이해진다. 따라서 대상 함정에 속하는 모든 피아식별기 연동 SW들이 수행해야 하는 요구사항과 그에 대한 기능 범위를 식별하고 공통된 특성을 취합하여 Table 4와 같이 나열하였다.

Table 4. Requirements and Functions of Existing IFF Interface SW

Requirement	Function
Connection with IFF	-Network Data Communication with IFF(TCP/UDP/etc.) -Interconversion IFF Msg
IFF Status Report Processing	Availability and Fault Info. Management of IFF Components
Ownship Info Transmission	Transmission Ownship Info. to IFF
IFF Interrogator Track Report Processing	IFF Track Info. Management (Track Creation, Modification, Deletion)
IFF GFE Control and Status Processing	IFF Control and Status Management of Interrogator/Transponder

기존의 피아식별기 연동 SW의 클래스 구조는 크게 통신 처리를 하는 클래스(Common GFE Manager, DDS 관련 클래스들)와 메시지 처리를 하는 클래스(GFE/CMS Manager)로 구분할 수 있다. GFE 측은 Baseline 2부터 통신 처리용을 위한 별도 Common GFE 클래스를 분리했으며, CMS 측은 미들웨어와 연관된 통신 클래스들을 자동 생성하여 사용하였다.

연동 SW의 표준성을 위해서는 함정과 장비의 성능에 따라 요구사항을 조정할 수 있으며, 그에 따른 기능 여부 또한 관리할 수 있어야 한다. 하지만 기존 피아식별기 연동 SW의 기능 구현 상황을 분석한 결과, 절차지향 형식에 가까운 클래스 분류법을 적용하여 향후의 가변요소에 대한 수정용이성을 보장할 수 없는 상황임을 확인하였다. 가장 대표적인 Baseline 2.0 기준의 기존 피아식별기 연동 SW에 대한 Class Diagram 설명은 아래 Fig. 3 및 Table 5와 같다.

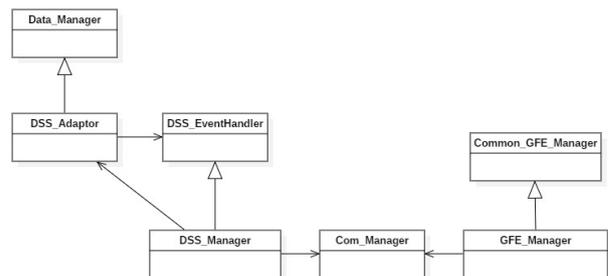


Fig. 3. Existing Class Diagram of IFF Interface SW in Baseline 2.0

Table 5. Existing Class Description of IFF Interface SW in Baseline 2.0

Class Name	Description
Common GFE Manager	Interface Manager of Common GFE (for Baseline 2.0)
GFE Manager	Interface Manager of GFE - Depends on GFE Info. (ex.GFE Type, Protocol, etc.)
DSS Manager	Interface Manager of CMS - Depends on CMS Info. (ex.middleware, Msg, etc.)
ComManager	Msg Communication Manager between GFE and CMS Class
Data Manager / DSS Adaptor / DSS EventHandler	CMS middleware Function Processing (Auto Gen.) - Depends on CMS middleware

**Step 2. First Architecture Design with NSCP**

이전 단계에서 식별된 기능을 기준으로 기존의 절차지향형 클래스 구조를 NSCP 기반의 객체지향형 클래스 구조로 재설계하는 과정을 진행하였다. NSCP의 주요 4가지의 공통 컴포넌트들과 연관된 하위 컴포넌트들을 표준화 설계하였다. 다만 이 과정에서 공통 컴포넌트 중 일부의 항목의 추가적인 변동성이 예상되었다. 따라서 다음 단계에서 해당 요소들의 가변성을 해소한 후에 최종적인 Class Diagram을 제안하도록 한다. 본 단계로 도출된 클래스 목록은 크게 Table 6의 CMS 환경, Table 7의 GFE 환경, Table 8의 내부 기능 부분으로 나누어 작성하였다.

Table 6. New Class Description of IFF Interface SW related CMS

Class Name	Description
CMS Interface Manager	Interface Manager of CMS
Data Manager	CMS middleware Function Processing (Auto Gen.)
Callback Handler	CMS middleware Callback Processing (Auto Gen.)
CMS Msg/Info Maker	CMS Data Converting between Msg(Ex) and Info.(In)

해당 CMS 설계는 기존 연동 SW에서 DSS Manager, Data Manager, DSS Adaptor, DSS EventHandler의 클래스들로 이루어진 설계를 대체한다. 대부분의 기존 체계 클래스들이 가진 미들웨어의 의존성을 최대한 분리하여 Data Manager/Callback Handler 2개로 축소하였으며, 두 개의 미들웨어에 따른 공용화 설계가 가능하도록 클래스 형식을 표준화하였다. 해당 클래스에 대한 송수신 접근의 혼재도 CMS Interface Manager에서만 단독 수

행하도록 이관하여 불필요해진 접근 클래스 DSSAdator를 삭제하였다. 또한 기존 CMS Manager에서 처리하던 체계 송수신 메시지의 변환 기능을 Msg 및 Info Maker 클래스 기능으로 이관하고 실제 기능 처리는 각 유관 SubFunction 클래스로 이관함으로써 CMS Interface Manager 클래스에게 단일책임을 갖게 했다.

Table 7. New Class Description of IFF Interface SW related GFE

Class Name	Description
GFE Interface Manager	Interface Manager of GFE
PSCInterface Manager / TCP, UDP, Serial Interface Manager	GFE middleware Function Processing
GFE Msg/Info Maker	GFE Data Converting between Msg(Ex) and Info.(In)

해당 GFE 설계는 기존 연동 SW에서 Common 또는 일반 GFE Manager 클래스를 대체한다. Baseline 별 불규칙하게 정의된 GFE의 통신 처리 클래스들을 표준화된 클래스들로 통일하였으며, CMS 설계와 동일하게 GFE 측면에서 메시지 변환 처리용 Maker 및 GFE Interface Manager 클래스를 표준화하여 적용하였다.

Table 8. New Class Description of IFF Interface SW related Functions

Class Name	Description
Function Manager	Manager of SubFunctions, Mediator between External Data and Internal Data
GFE Status Manager	SubFunction Manager, Gfe Status Management of IFF
IFF ModeInfo Manager	SubFunction Manager, IFF Identification Mode Info. Management
Interrogator Control	SubFunction Manager, IFF Interrogator Control and Status Management
Transponder Control	SubFunction Manager, IFF Transponder Control and Status Management
OSD Info Manager	SubFunction Manager, Ownship Info. Management
Primitive Track Manager	SubFunction Manager, IFF Track Report and CMS Primitive Track Management
Radar Mode Manager	SubFunction Manager, CMS Radar Mode Info. Management For IFF
Radar Selector	SubFunction Manager, IFF Radar Selection Control and Status Management

해당 Function 설계는 기존 연동 SW에서 GFE/CMS Manager에 통합되어 있던 실제 피아식별기 처리 기능 부분을 각 유의미한 단위의 기능들로 분리하여 대체 설계하였다. 이를 통해 대상 함정 및 장비에 따라 기능 여부를 관리할 수 있게 되고, 추후 기능의 확장도 용이하다. 이는 연동 SW의 표준성을 지향하는 구조이다.

**Step3. Analysis with Feature Model and Variable Domain Classification**

앞서 1차 설계한 피아식별기 연동 SW의 기능을 휘처 모델에 적용하였다. Fig. 4는 휘처 도형을 통해 해당 연동 SW의 기능을 휘처 모델로 구성한 그림이다.

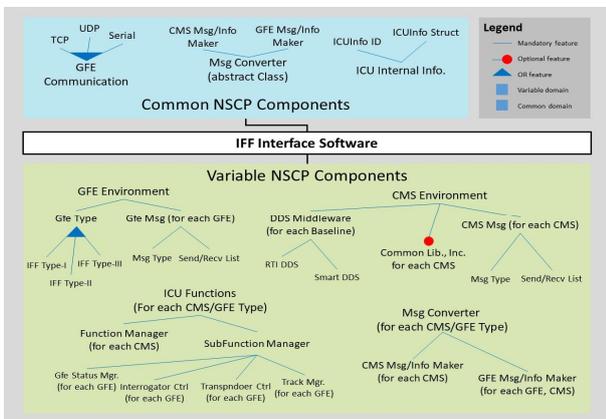


Fig. 4. Feature Model of IFF Interface SW

- Common domain : 체계 및 장비 연동 정보 관련 피아식별기 연동 소프트웨어의 가변요소 식으로는 GFE 연동 시 통신 유형 별 처리 기능, Msg Converter 관련 공통 기능, 연동단 내부 공통 정보가 포함되었다.
- Variable domain : GFE 타입 정보 및 그에 따른 송수신 메시지 정보와 같은 GFE 개별 환경 정보와, DDS 미들웨어 종류 및 각 체계 별로 상이한 송수신 메시지, 공통 라이브러리 등을 포함하는 CMS 개별 환경 정보가 포함되었다. 이와 더불어 해당 GFE/CMS의 개별 타입에 따라 변동되는 외부 메시지와 내부 정보 간 상호 변환 기능이 포함되었고, 변환을 거쳐 수신된 모든 내부 정보 목록을 전달 및 관리하기 위한 Function Manager와 실제 처리 역할을 하는 SubFunction Manager 항목 중 개별 GFE 단위로 기능 처리가 필요한 대상 기능 관리자들도 포함되었다.

피아식별기 연동을 위한 표준 SW는 탑재되는 개별 함정체계와 장비의 특성에 따라 다양한 처리 방식을 지원해야 하고, 그에 따라 필요한 정보를 제공할 수 있도록 해당 가변성을 공용화 및 표준화하여 관리하여야 한다.

**Step 4. Final Architecture Design with Feature Model and S.O.L.I.D Principle**

휘처모델의 가변요소로 식별된 항목들을 위한 클래스를 추가하였다. 새로운 클래스를 추가하면서 클래스 작성의 5대 설계 원칙인 S.O.L.I.D[16-17]를 준수하고자 노력하였다. 설계 완료한 Class Diagram에 대해 Fig. 5의 CMS 설계, Fig. 6의 GFE 설계, Fig. 7의 Function 설계 단위로 확인할 수 있으며, 이전 단계 대비 변화된 클래스와 관련된 설명을 함께 서술하였다. 최종적으로 제안하는 아키텍처에 대한 전체 Class Diagram은 Fig. 8과 같다.

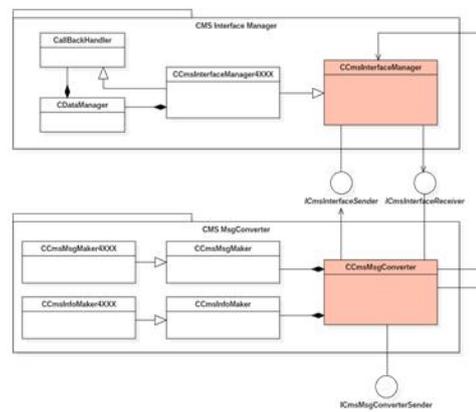


Fig. 5. CMS Design of New IFF Standard Interface SW

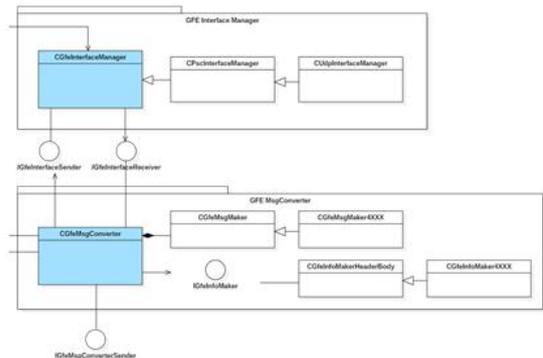


Fig. 6. GFE Design of New IFF Standard Interface SW

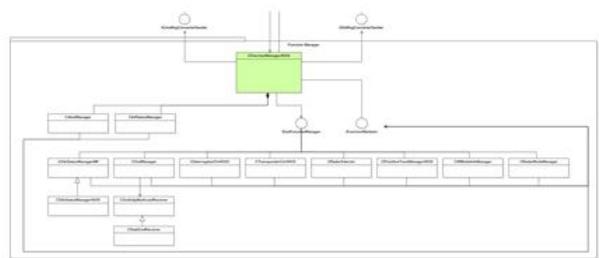


Fig. 7. Function Design of New IFF Standard Interface SW

● 개별 함정 단위 클래스

- CMS Interface Manager, CMS Msg/Info Maker : 대상 클래스를 개별 함정을 기준으로 하위 클래스로 분류하였다. 함정 별 상이한 CMS 메시지 목록에 대해 메시지 변환 및 송수신 처리를 보장한다.

- Function Manager : 개별 함정 별 상이한 CMS 및 GFE 메시지 목록에 대해 적절한 Maker 또는 SubFunction Manager로의 전달 처리를 보장한다. 함정 별 보유한 기능에 따라 가변적인 SubFunction Manager 그룹을 생성하는 것을 보장한다.

● 개별 미들웨어 단위 클래스

- Data Manager / Callback Handler : 두 가지 미들웨어와 연관된 상용 API를 표준적으로 처리할 수 있도록 외부 호출 함수 구조를 표준화하였다. 사용자 측면에서 미들웨어와 무관하게 공통된 메시지 송수신 기능을 보장한다.

● 개별 장비 단위 클래스

- GFE Interface Manager, GFE Msg/Info Maker : 대상 클래스를 연관 GFE Type을 기준으로 하위 클래스로 분류하였다. 장비 타입 별 상이한 GFE 메시지 목록에 대해 메시지 변환 및 송수신 처리를 보장한다. 여기서 기식 별된 GFE의 공통 특성에 맞춰 <Type-I, II>를 하나의 상위 클래스로 생성하고, <Type-III>를 또 다른 상위 클래스로 생성한다. 탑재 함정의 레이더 정보 등 개별 함정 단위의 정보 전달을 위해 그에 대한 함정 별 하위 클래스를 생성하여 사용한다.

- SubFMs(GFE Status Manager, Interrogator Control, Transponder Control, Primitive Track Manager) : 해당 하위 기능관리자 클래스들은 장비의 특성에 따라 동작 개념이 달라질 수 있기 때문에, 장비 타입 별 표준화 클래스를 생성하여 사용한다.

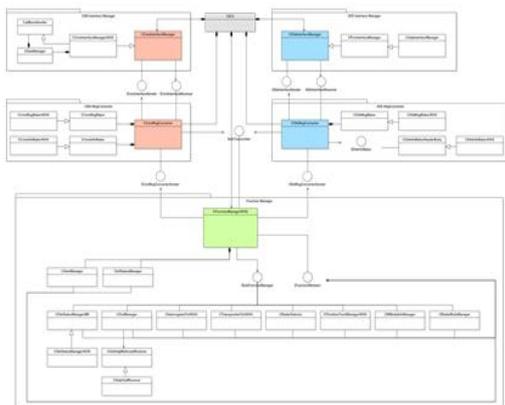


Fig. 8. New Class Diagram of IFF Standard Interface SW With Feature Model and SOLID Principle

본 논문에서는 다중 함정 및 다중 장비 타입에 대해 개별 구현된 피아식별기 연동 SW를 하나의 표준화된 피아식별기 연동 아키텍처를 설계 및 구현하는 연구를 수행하였고, 기대효과는 다음과 같다.

- 다중 함정에 탑재된 개별 연동 SW들을 표준 형식의 공통 아키텍처로 설계함으로써 CMS 소프트웨어의 관리 효율성 증대
- 체계 및 장비 연동 정보 관련 피아식별기 연동 소프트웨어의 가변요소 식별을 통한 소스코드의 재사용성 및 확장성 증대
- 피아식별기 연동 소프트웨어 신뢰성 시험 범위 최소화를 통한 개발비용 감소

#### IV. Software Evaluation

본 논문에서 함정전투체계 피아식별기 표준 연동 아키텍처를 적용한 소프트웨어의 개선 정도를 확인하기 위해 체계 혹은 장비 변경에 대한 수정 요소와 신뢰성 시험 수행의 총 시간을 비교하여 평가하였다. 평가에 대한 실험 환경은 아래 Fig. 9 및 Table 9와 같다.

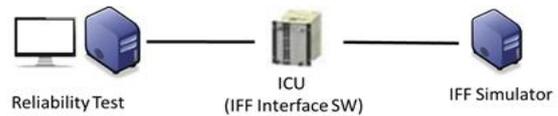


Fig. 9. Test Environment for Verification

Table 9. Performance of Test Environment

Item	Reliability Test	ICU	IFF Simulator
CPU	Intel Core i9-10900 @3.70GHz	Intel @2.40GHz	Intel Core i7-11700 @2.50GHz
Memory	96 GB	4 GB	32 GB
OS	Windows 10	RTST Linux	Windows 10

제한한 피아식별기 연동 표준 아키텍처와 기존의 피아식별기 연동 소프트웨어의 평가를 위해 체계 및 장비 정보 변경에 대한 의존성을 비교하는 실험을 진행하였다. 가장 많은 개발 비용이 투입되는 소스코드 수정에 대해, 수정이 발생하는 대상 클래스의 수와 변경 라인 수를 측정하고 재사용률을 도출해 비교하였다. 또한, 본 과제의 피아식별기 연동 SW는 최신 Baseline의 엄격한 신뢰성 시험을 기준을 충족해야 하므로 본 환경도 동일한 시험기준으로 진행

하였다. 정적분석은 CodeScroll Inspector, SNIPER 도구, 동적분석은 Cover[18] 도구를 사용하였으며, 실제 신뢰성시험을 수행한 시간을 측정하고 비교하였다. 실험 시나리오는 IFF Mode-5 성능개량 대상 피아식별기 연동 SW의 변경 CASE 중 Best와 Worst의 2가지 CASE가 연달아 발생할 때로 설정하였다.

- CASE-1 : 가장 구형의 체계(Baseline 1.0)에서 신규 장비(Type-I)로 HW를 교체하는 경우
- CASE-2 : 가장 최신의 체계(Baseline 2.0)에서 기존 장비의 SW를 업그레이드(Type-III)하는 경우

**CASE-1 : CMS Baseline 1.0, GFE Type-I (HW replace)**

Table 10. Analysis of Code Reuse Rate (CASE-1)

Analysis Item	Exist SW	New SW
Total number of classes( $\alpha$ )	5	45
Number of common classes( $\beta$ )	0	35
Number of modified classes according to changes in CMS( $\gamma$ )	4	10
Number of modified classes according to changes in GFE( $\delta$ )	5	10
Common class rate( $\beta/\alpha$ )	0 %	78 %
Structural change rate according to changes in CMS( $\gamma/\alpha$ )	80%	22 %
Structural change rate according to changes in GFE( $\delta/\alpha$ )	100 %	22 %

Table 11. Compare time to perform Reliability Test (CASE-1)

Reliability Test	Exist SW	New SW	Diff.
Static Test	75 hours	32 hours	-43 hours
Dynamic Test	90 hours	40 hours	-50 hours

CASE-1의 경우 가변성이 높은 구형 체계를 대상으로 하기 때문에 Table 10과 같이 기존 SW 대비 제안 SW의 전체 클래스 개수는 증가하였으나 공용 클래스 포함 비율이 기존 대비 78% 향상되어 높은 유지보수성을 확보했다. 또한, 대상 가변 정보의 전체 변경에 대한 구조 변경 비율은 체계 측면에서 58%, 장비 측면에서 78%의 감소하여 높은 재사용성을 확보했다. 신뢰성 수행 시간 비교의 경우, Table 11과 같이 정적 분석 시 최대 43시간, 동적 분석 시 최대 50시간이 절약됨을 확인하였다. 이는 공용 클래스의 최초 수행시간을 포함한 것으로, 이후 CASE-2에서는 해당 시간을 제외하여 신뢰성의 효율성을 추가 판단한다.

**CASE-2 : CMS Baseline 2.0, GFE Type-III(SW Upgrade)**

Table 12. Analysis of Code Reuse Rate (CASE-2)

Analysis Item	Exist SW	New SW
Total number of classes( $\alpha$ )	6	41
Number of common classes( $\beta$ )	1	32
Number of modified classes according to changes in CMS( $\gamma$ )	4	9
Number of modified classes according to changes in GFE( $\delta$ )	3	8
Common class rate( $\beta/\alpha$ )	17 %	78 %
Structural change rate according to changes in CMS( $\gamma/\alpha$ )	67%	22 %
Structural change rate according to changes in GFE( $\delta/\alpha$ )	50 %	20 %

Table 13. Compare time to perform Reliability Test (CASE-2)

Reliability Test	Exist SW	New SW	Diff.
Static Test	40 hours	14 hours	-26 hours
Dynamic Test	70 hours	25 hours	-45 hours

CASE-2의 경우 가변성이 낮은 최신 체계임에도 Table 12와 같이 기존 SW 대비 제안 SW의 공용 클래스 포함 비율이 기존 대비 61% 향상되어 높은 유지보수성을 유지했고, 대상 가변 정보의 전체 변경에 대한 구조 변경 비율 또한 체계 측면에서 45%, 장비 측면에서 30% 감소되어 재사용성 또한 여전히 향상된 수치임을 확인할 수 있었다. 신뢰성의 경우 Table 13과 같이 정적 분석 시 최대 26시간, 동적 분석 시 최대 45시간이 절약됨을 확인하였다. CASE-2에서는 특히 신뢰성 수행 시간 비교 시 이전 CASE-1의 새로운 SW와의 비교에서도 정적 분석 시 18시간, 동적분석 시 15시간의 확연한 수행 차이를 보인다. 그 이유는 CASE-1에서 이미 수행한 공용 클래스들은 표준 SW 간 결과 공유가 가능하기 때문에, 이를 제외한 개별 항목만 수행 시간에 포함되기 때문이다. 이러한 격차는 CASE-2 -> CASE-1 순으로 시나리오 적용 시 보다 큰 절감 효과를 예상할 수 있다. 또한, 본 제안 과제의 목적과 같이 많은 대상 함정 및 장비에 대해 복잡한 가변 CASE들을 반복 비교할수록 기존 대비 표준 연동 SW의 효율성은 극대화될 수 있을 것이라 기대한다.

## V. Conclusions

본 논문에서는 다수의 함정 전투체계에 탑재된 기존 피아식별기 연동 소프트웨어들을 효과적으로 대체하는 표준 연동 아키텍처 연구를 진행하였다. 제안된 피아식별기 연동 표준 아키텍처는 차기 함정전투체계의 표준 연동 아키텍처(NSCP)를 바탕으로 하여 선진화된 표준 연동 설계를 적용하였으며, 휘처 모델링을 통해 다중 체계 및 장비 연동에 대한 가변요소를 추가 식별 및 재설계함으로써 표준 아키텍처를 최종 설계하였다.

새로운 표준 연동 SW는 공용 클래스 포함 비율, 체계 및 장비 정보 변경 시 구조 변경 비율과 신뢰성시험 수행 시간 비교를 통해 소스코드의 변경 용이성과 재사용성의 향상으로 개발비용의 감소를 확인하였다. 제한된 자원에서 고품질의 기술성을 발휘하는 것은 해군 함정에서 매우 중요하므로, 기존 함정에 대한 SW 기반의 성능개량 사업은 더욱 주요한 과제로 자리할 것이다. 따라서, 본 논문에서 제안한 방법은 차후 도래할 다양한 함정전투체계의 성능개량 과제를 위한 연구 자료로 효과적으로 활용될 것이다.

## REFERENCES

- [1] S. C. Noh, et al., "SW Reliability Test Verification in PerformanceImprovement Project: An Efficiency Enhancement Scheme", *Journal of the Korea Academia-Industrial cooperation Society* Vol. 23, No. 7, pp. 389-398, July 2022. DOI: 10.5762/KAIS.2022.23.7.389
- [2] MISRA, <http://www.misra.org.uk/>
- [3] H. Y. Shin, et al., "A Study on the Establishment of Risk Assessment Criteria for the Dynamic Reliability Test of Performance Improvement Weapon System using Recycling Software", *Journal of KIISE*, Vol. 49, No. 1, pp. 67-77, January 2022. DOI: 10.5626/JOK.2022.49.1.67
- [4] J. Y. Park, et al., "A Study on IISS Software Architecture of Combat Management System for improving modifiability", *Journal of The Korea Society of Computer and Information*, Vol. 25, No. 2, pp. 130-140, May 2020. DOI: 10.9708/jksci.2020.25.02.133
- [5] S. M. Kwon, S. M. Jung, "Virtualization based high efficiency naval combat management system design and performance analysis", *Journal of The Korea Society of Computer and Information*, Vol. 23, No. 11, pp. 9-15, November 2018. DOI: 10.9708/jksci.2018.23.11.009
- [6] D. K. Kim, et al., "A Study on Efficient Design of Surveillance RADAR Interface Control Unit in Naval Combat System", *Journal of The Korea Society of Computer and Information*, Vol. 28, No. 11, pp. 125-134, November 2023. DOI: 10.9708/jksci.2023.28.11.125
- [7] H. J. Choi, "A Study on the Software Standardization and Simulator Design for Efficient Reliability Test in Combat System", *Journal of The Korea Society of Computer and Information*, Vol. 27, No. 12, pp. 151-159, December 2022. DOI: 10.9708/jksci.2022.27.12.151
- [8] DDS, <https://www.omg.org/omg-dds-portal/>
- [9] Y. D. Heo, "A Study on the Standardization of System Support Software in the Combat Management System", *Journal of the Korea Society of Computer and Information*, Vol. 25, No. 11, pp. 147-155, November 2020. DOI: 10.9708/jksci.2020.25.11.147
- [10] Y. H. Jeong, et al., "The Design of a Multi-Function Radar Simulator for the Identification of Friend or Foe(IFF) in the Mode-5 Product Improvement Program", *Journal of KIISE*, Vol. 47, No. 6, pp. 622-628, June 2020. DOI: 10.5626/JOK.2020.47.6.622
- [11] C. S. Baek and J. H. Ahn, "A Study of the Standard Interface Architecture of Naval Combat Management System", *Journal of the Korea Society of Computer and Information*, Vol. 26, No. 1, pp. 147-154, January 2021. DOI: 10.9708/jksci.2021.26.01.147
- [12] K. M. Yang, et al., "Modeling FORM Architectures Based on UML 2.0 Profiling", *Journal of KISS:Software and Applications*, Vol. 36, No. 6, pp. 431-442, June 2009.
- [13] C. Y. Song, et al., "A Formal Specification and Checking Technique of Feature model using Z language", *Journal of The Korea Society of Computer and Information*, Vol. 18, No. 1, pp. 123-136, January 2013. DOI: 10.9708/jksci.2013.18.1.123
- [14] CMS of hanwhasystems, [https://www.hanwhasystems.com/kr/business/defense/naval/combat\\_index.do](https://www.hanwhasystems.com/kr/business/defense/naval/combat_index.do)
- [15] Smart DDS of hanwhasystems, <https://www.hanwhasystems.com/kr/business/defense/naval/combat02.do>
- [16] Robert C. Martin, "Agile Software Development, Principles, Patterns, and Practices," Prentice Hall, New Jersey, pp. 95-145, 2002.
- [17] Robert C. Martin, "Clean Code: A Handbook of Agile Software Craftsmanship," Prentice Hall, New Jersey, pp. 138-140, 2008.
- [18] Cover, [https://www.suresofttech.com/solution/code\\_tool.php#cover/](https://www.suresofttech.com/solution/code_tool.php#cover/)

## Authors



Yeon-Hee Noh received the B.S. degree in Computer Engineering from Kyungpook National University, Korea. She is currently working in Hanwha Systems Co. from 2016. She is interested in Naval Combat

Management System, Interface Control Unit Software, Software Reuse and so on.



Dong-Han Jung received B.S degree in Electronics from Kyungpook National University, Korea. He is currently working in Hanwha Systems Co. from 2009. He is interested in Unmanned System Software,

Human Computer Interface and System Engineering.



Young-San Kim received B.S degree in Computer Science and Electrical Engineering from Handong University, Pohang, Korea. He is currently working in Hanwha Systems Co. from 2009. He is interest in Naval

Combat Management System Software, Human Computer Interface and System Engineering.



Hyo-Jo Lee received B.S degree in Electronics from Kyungpook National University, Korea. He is currently working in Hanwha Systems Co. from 2010. He is interest in Naval Combat System and Combat

System Support Software.