Case study – e-Navigation trials on APPS project

e-Navigation Underway International 2016

2 – 4 February 2016

Kilyong Kim (GMT Co., Ltd.)

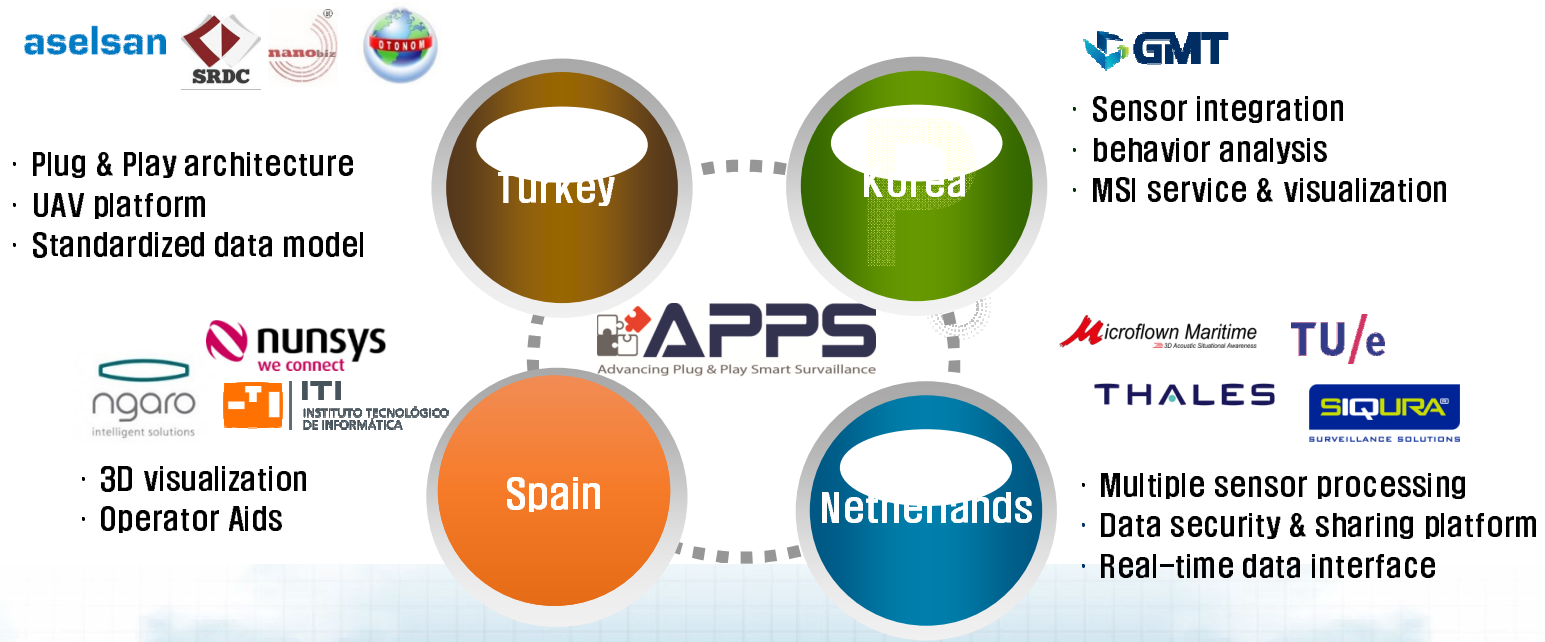
Co-author : Seojeong Lee (Korea Maritime and Ocean University)

Contents

I. Introduction of APPS Project

APPS(Advancing Plug & Play Smart Surveillance) Project

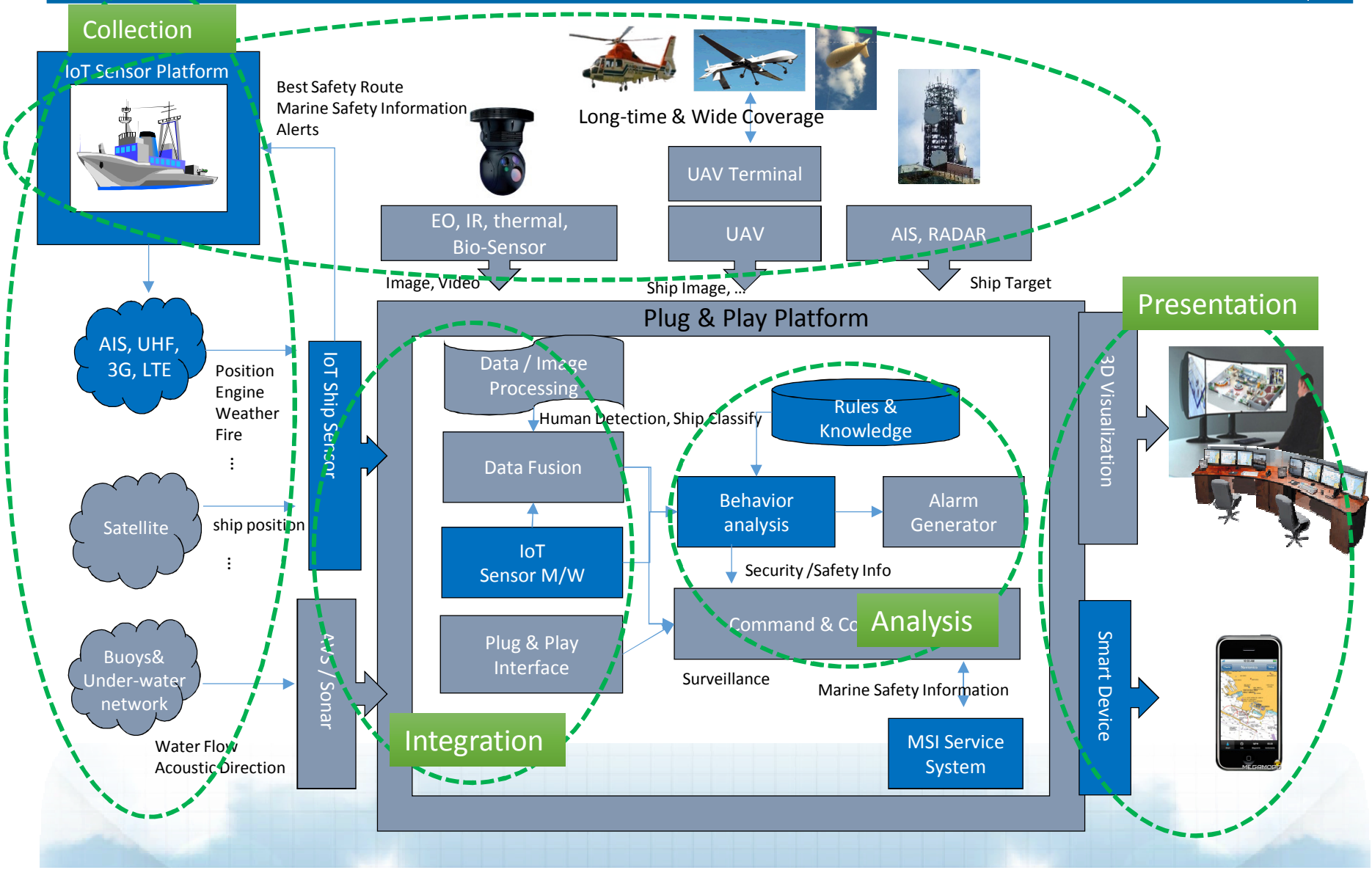
- Eureka R&D project
  - From Jan 2015 to Dec 2017
  - 15 partners from Netherlands, Turkey, Spain, Republic of Korea
  - 2 end users (Port of Rotterdam, Republic of Turkey Ministry of Transport, Maritime Affairs and Communication)



APPS(Advancing Plug & Play Smart Surveillance) Project

- Purpose
  - Enable the development of **plug & play** solutions
  - Enhance sensors-processing and intelligent decision-making capabilities and intelligent operator-aids of such systems to achieve **smart surveillance**
  - Developing acoustic and physico-chemical sensors, LTA(Lighter Than Air) and stationary wing UAV(Unmanned Aerial vehicle)

APPS System concept



Ⅱ. e-Navigation trials on APPS
(from Jan to Dec 2015)

Case #1. How to collect sensor data on board

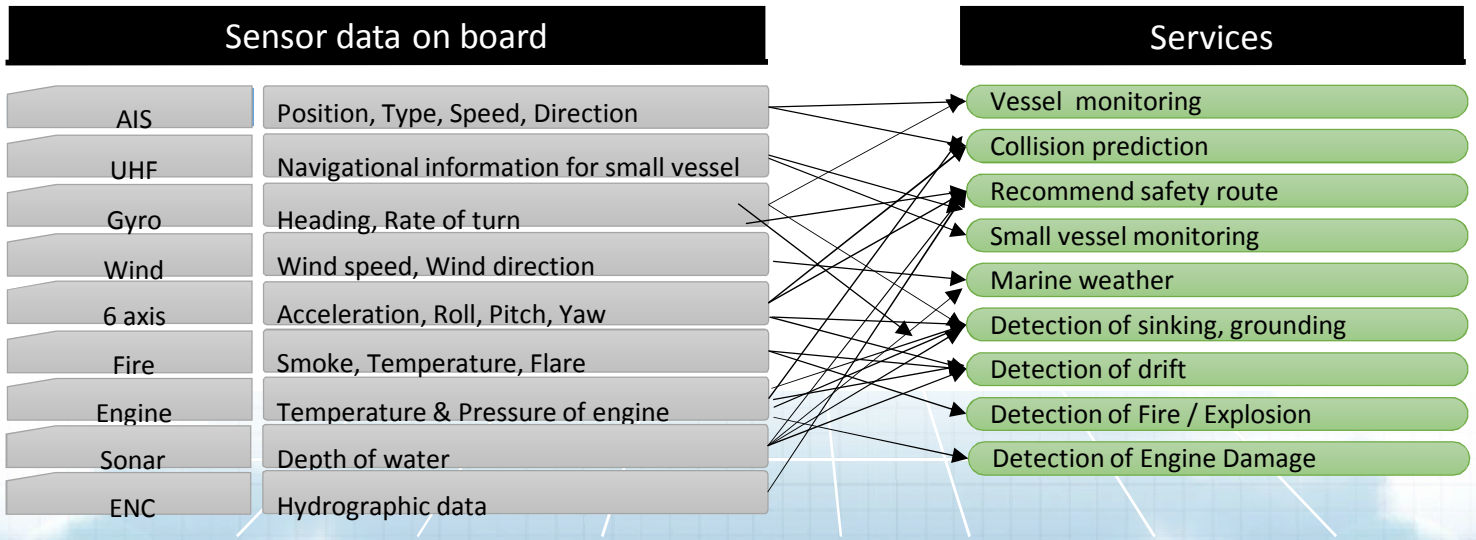
Case #2. Detection and tracking of small or non-cooperative vessels

Case #3. How to improve software quality

Case #1. How to collect sensor data on board

Data collection from sensors on board

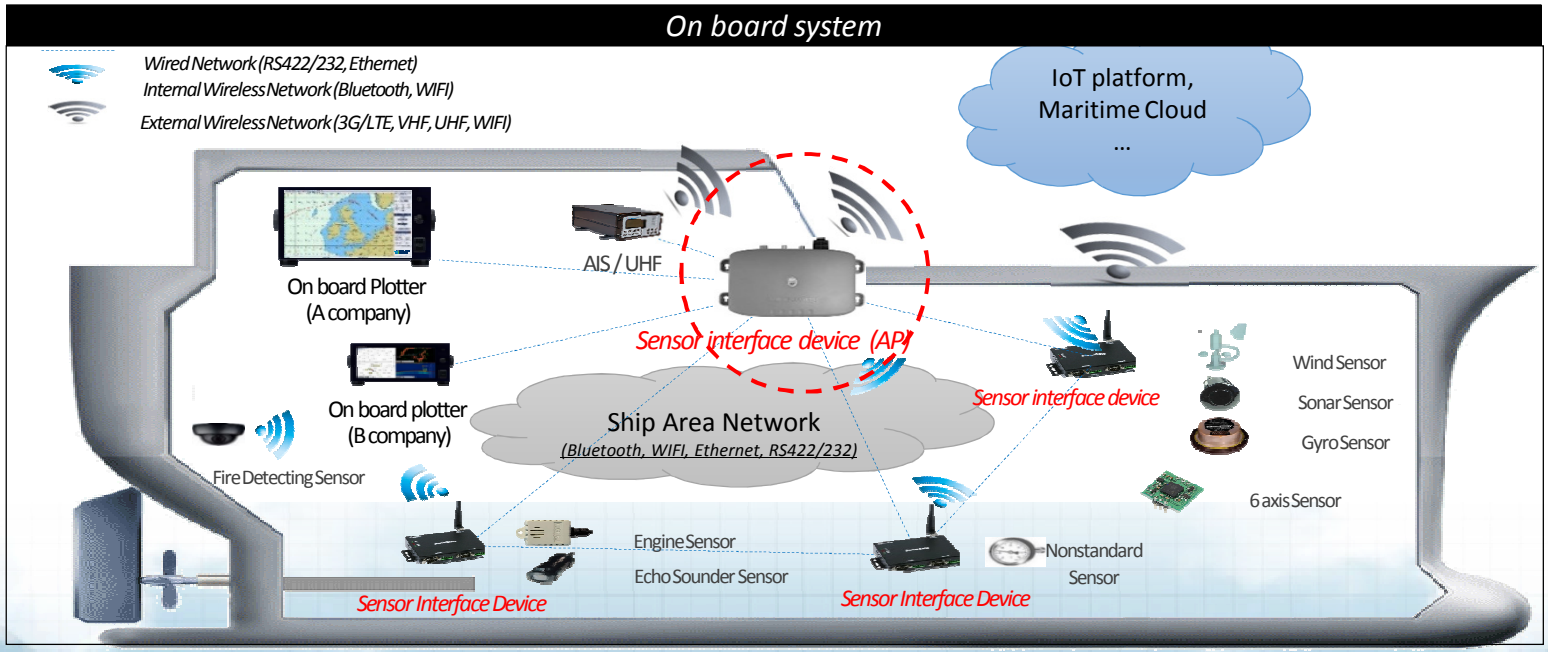
- Background
  - There is various accident risk in maritime, such as collision, grounding, sinking , fire/explosion and engine damage
  - There are several devices and sensors that are installed on vessels in compliance with international regulations.
  - Sensor data on board is key parameters for behavior analysis to predict maritime accident and to detect abnormal behavior.



Case #1. How to collect sensor data on board

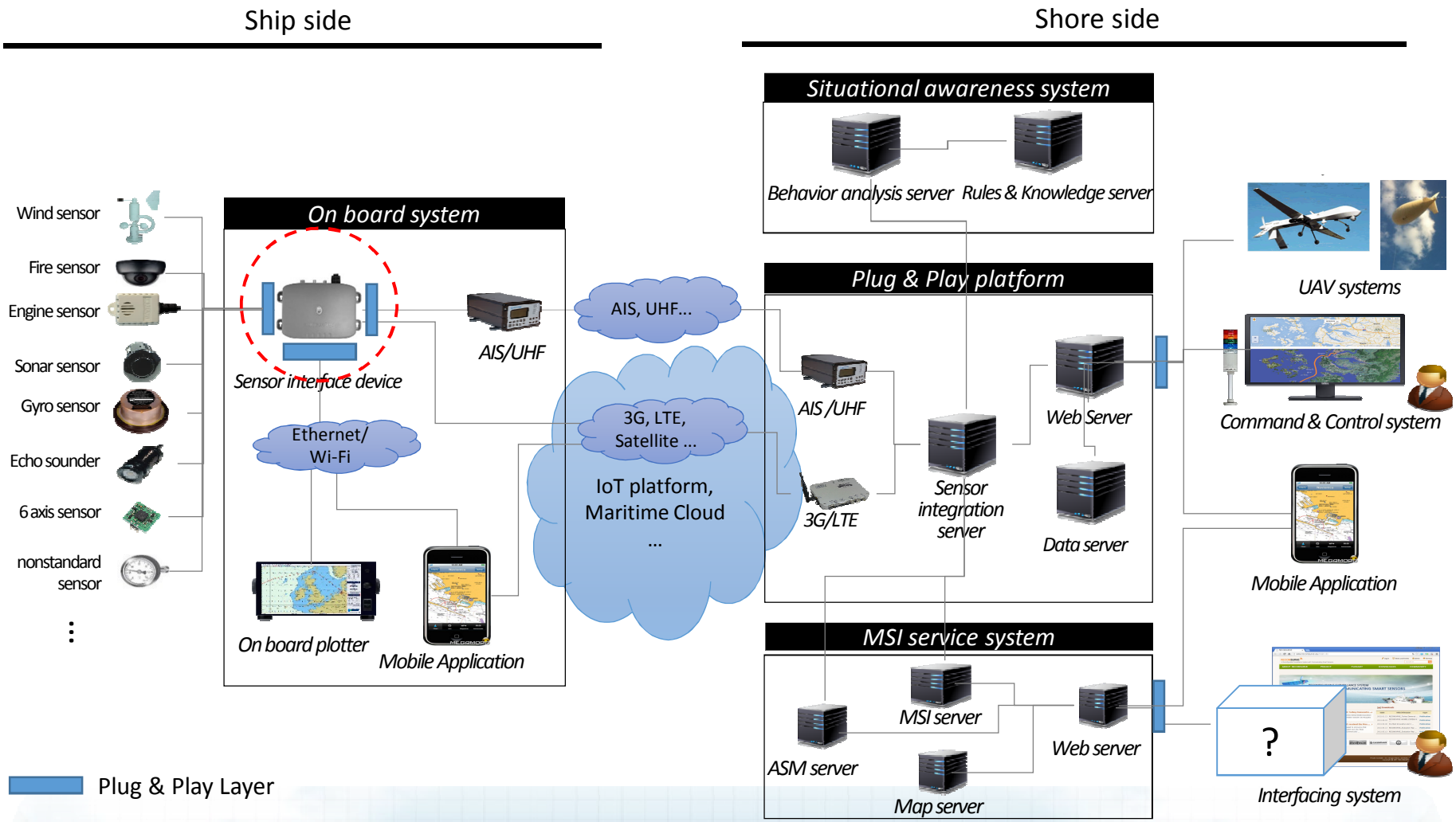
Requirements for Sensor interface device

- To interface with legacy sensors on board
  - Need to apply international standard (IEC61162-1,2,3,450, SensorML...)
- Algorithm to select optimal maritime communication autonomously
  - Considering type, priority, transmission interval of collected data and cost and data rate of available network
- Data exchange between ship and shore side system
  - Using proper interface with open API (IoT platform, Maritime Cloud...)



Case #1. How to collect sensor data on board

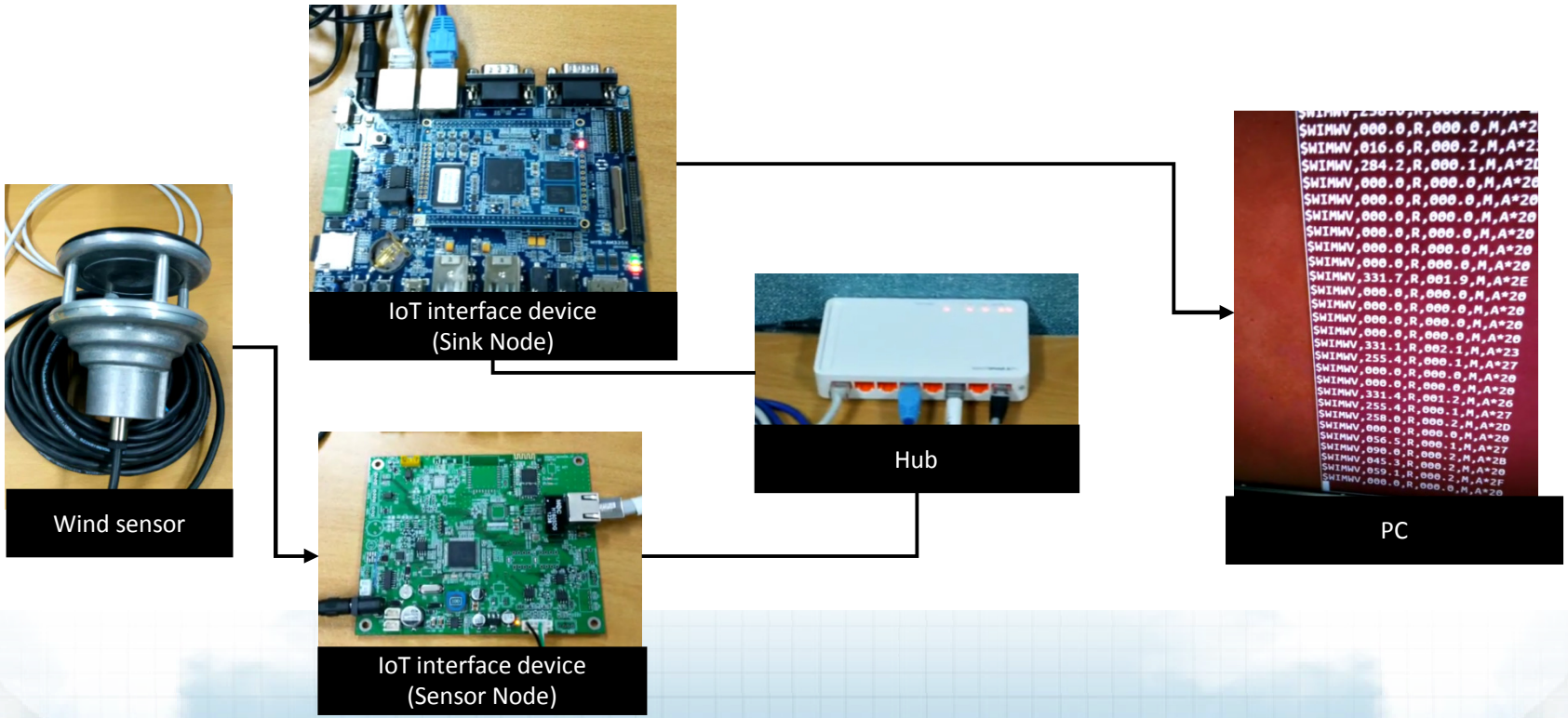
System configuration



Case #1. How to collect sensor data on board

Prototype of sensor Interface device

- We implemented the prototype of hardware to interface with some of the sensors on board.
- This year, we are going to implement the plug & play function and other remaining algorithm based on this prototype.



Case #2. Detection and tracking of small or non-cooperative vessels

Background

- Detection relies on radar & AIS signals (which non-cooperative vessels do not send).
- Existing surveillance systems based on radar only are not always able to recognize not reported threats and issues, such as non-cooperative vessels, carrying illegal immigrants.
- We are going to use multiple sensors to detect and classify the vessels.
  - IR(Infrared Ray) cameras, PTZ(Pan Tilt Zoom) cameras
  - Acoustic sensors
- Unmanned Aerial Vehicles (UAV)
  - Lighter-Than-Air
  - Fixed-wing UAV

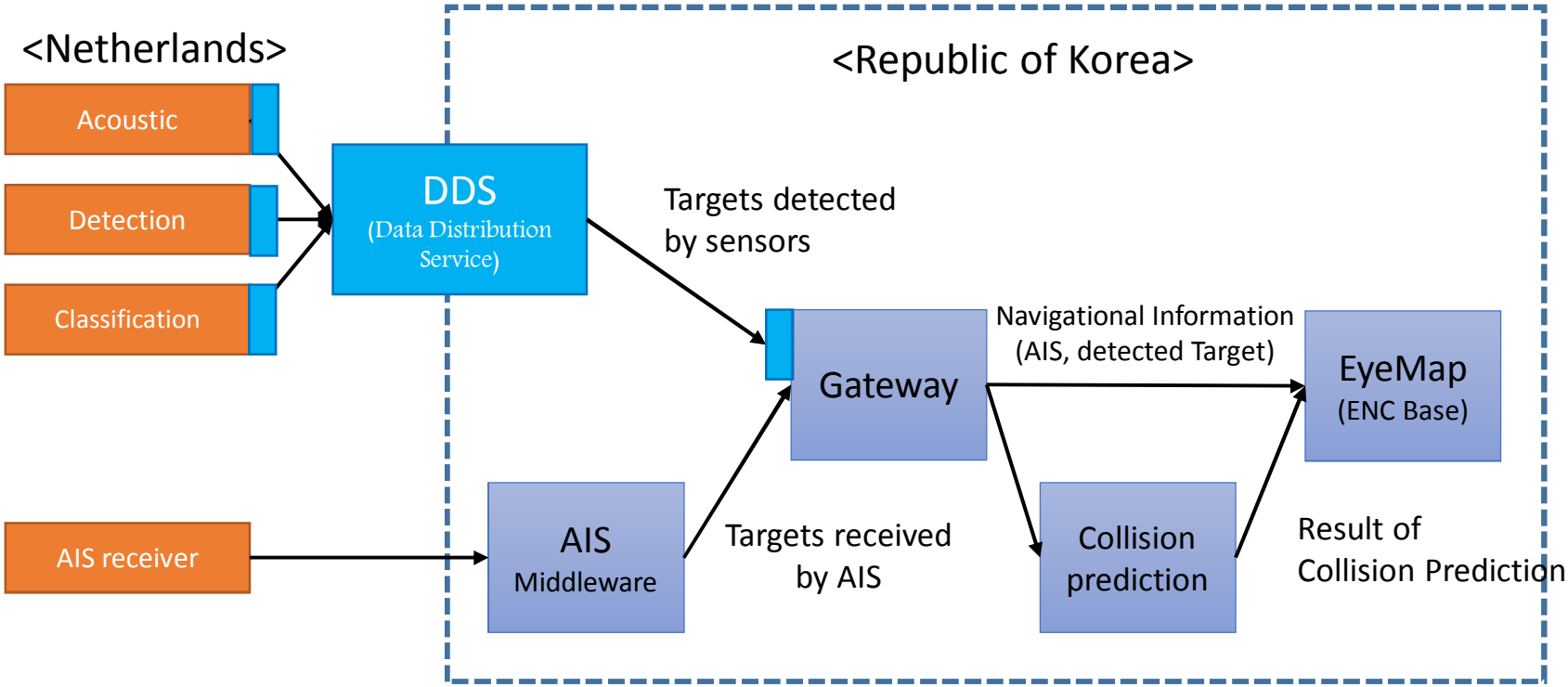
1st demonstration in port of Rotterdam

- The 1st demonstration in port of Rotterdam (17th Dec, 2015)
- Purpose of demonstration
  - To detect vessels using visual sensors
  - To integrate by DDS(Data Distribution Service) system
  - To portray detected targets and analyze for collision prediction based on ENC
- Installed cameras
  - 2x fixed thermal cameras
  - 1x Pan-Tilt-Zoom (PTZ) cameras
  - Extensions for 5x additional cameras

Case #2. Detection and tracking of small or non-cooperative vessels

1st demonstration in port of Rotterdam

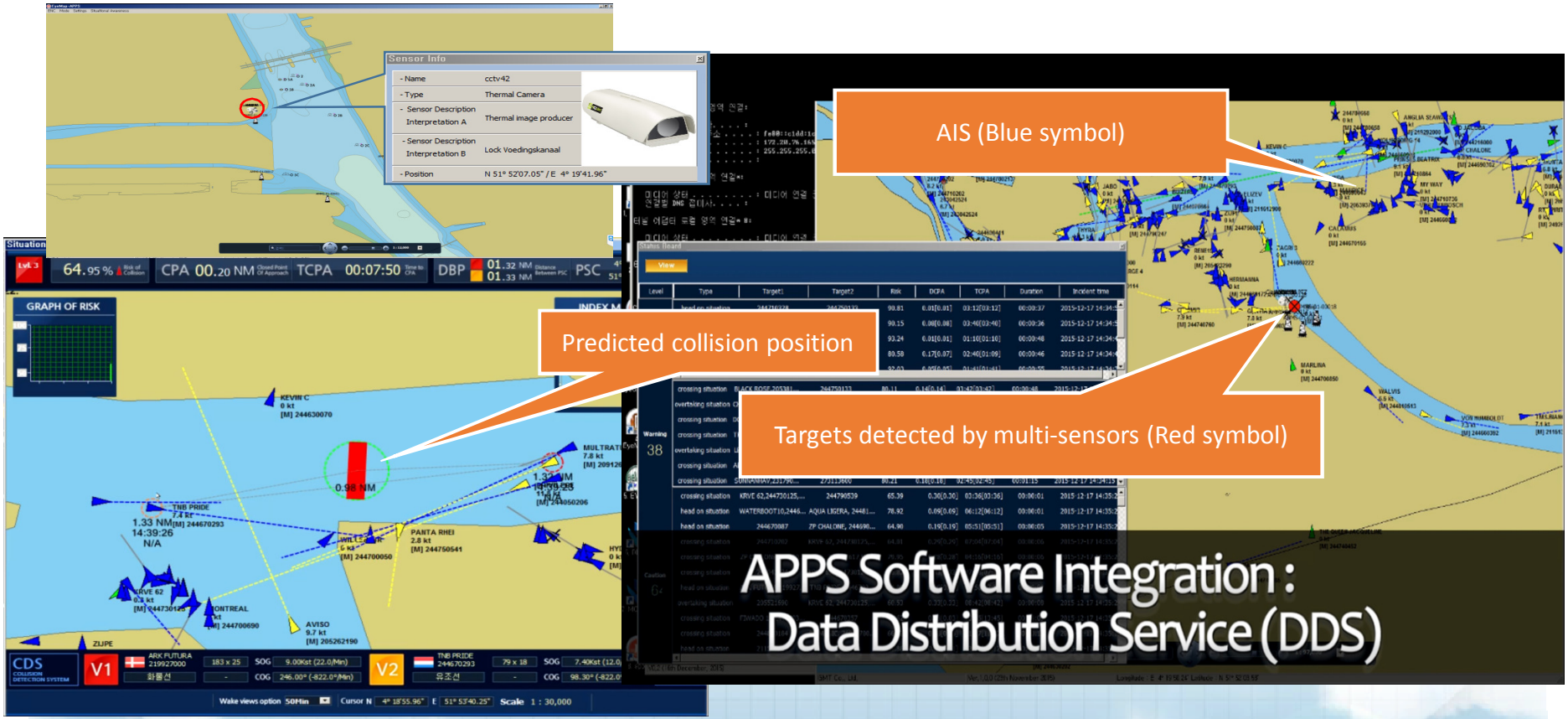
- Data integration using DDS(Data Distribution Service)
- Data of detected target : ID, position, SOG, COG, size of ship...



Case #2. Detection and tracking of small or non-cooperative vessels

1st demonstration in port of Rotterdam

- Integrated detected target using DDS
- Displayed target based on ENC (red symbol: detected by sensor)
- Collision prediction demonstrated using AIS and detected target data



Case #3. How to improve software quality

How to improve software quality

- Background
  - The guideline on "Software Quality Assurance and Human-Centered Design for e-Navigation "was adopted by IMO
  - In order to assure the software quality, it is necessary to follow a defined procedure throughout the entire software development period.
  - The utilization of tools to support this can become one of important factors to improve productivity of software development and to keep the software quality consistently.

Case #3. How to improve software quality

Adopted CI(Continuous Integration) tool

- Adopted CI(Continuous Integration) tool to keep software quality consistently
- Automated the process which compiles – tests – verifies – deploys source codes
- Four steps to apply CI tool

Adjusting software development process → Tailoring templates and documents → Adopting CI tools → Applying to a sub-system of APPS

SoftwareManager FinalBuilder

Improvement of

- Managing the status of progress **visually**
- **Communicating** between stakeholders **using templates and documents**
- **Tracing the history** of change of every requirements
- **Reducing cost and time** to integrate and build source codes
- Understanding the necessity of quality management

More than before


Ⅲ. Conclusion

Conclusion

- Introduced e-Navigation experience in EUREKA-supported APPS project
- Conducted 3 case studies focusing on e-Navigation.
- This result can be connected to the existing test-beds with proper interfaces in the future.

aselsan

THALES

TU/e

Thank you

Kilyong Kim
GMT Co., Ltd. yonjjang@gmtc.kr
Seojeong Lee
Korea Maritime and Ocean University sjlee@kmou.ac.kr