

IHO S-100 Working Group

KHOA S-100 Test bed

Presented by KHOA



General Information

- Name of testbed: KHOA S-100 Testbed Project
- Location of testbed: Busan, ROK
- Time and duration of testbed: Ongoing till S-100 testbed project of IHO is completed
- Contact person(s): Yong Baek, ybaek@korea.kr (Project Manager)
- Testbed website: N/A
- Organization(s) involved : KHOA (Korea Hydrographic and Oceanographic Agency)
- Funding program and budget: KHOA
- Last Edited/Updated: January 30, 2019



Executive summary

- Aims
 - to develop Phase 3 (Simple Viewer) and Phase 6 (shore based ECDIS) of S-100 test framework to support the S-100 Testbed project of the IHO
 - Through the development of Phase 3 (Simple Viewer), Phase 1, a catalogue produced through S-100 infrastructure, and Phase 2 including simple production tool, S-101 converted, S-100 simple overlay can be validated.
 - Phase 4 and Phase 5 can be validated through the development of Phase 6 (shore based ECDIS)



Methodology used for data collection

- Data applied to KHOA S-100 Testbed
 - S-10X Feature Catalogue (XML)
 - S-10X Portrayal Catalogue (XSLT, Lua)
 - TDS in 8211 (S-101)
 - TDS in GML (S-122, S-123, S-124, S-127, S-128)
 - TDS in HDF-5 (S-102, S-104, S-111)



Technical solutions used

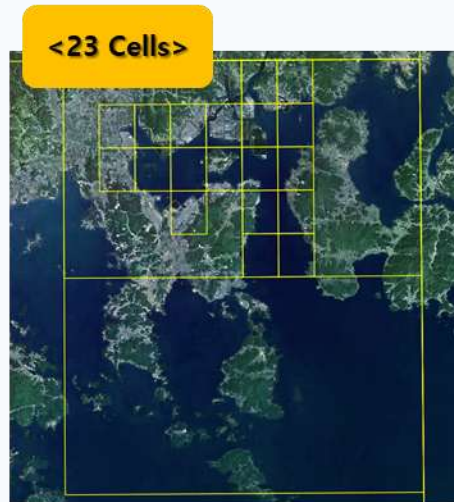
- The following solutions were applied
 - **S-100 Portrayal process**: Portrayal rule was applied and screen was presented (XSLT application method) about S-10X TDS according to S-100 Chapter 9.
 - **S-10X TDS**: Processing various data formats including 8211, GML and HDF-5
 - **S-100 Exchange Catalogue**: Loading and processing data according to information included in exchange set catalogue
 - **S-100 Interoperability**: Organizing screen presentation methods among S-100 product specifications from portrayal perspective
 - **Plug & Play Concept**: Changes to data model of S-100 product specifications are applied to Feature/Portrayal Catalogue. Data processing and screen presentation are applied according to Catalogues information.



Creation of S-100 test datasets

- S-101 ENC

↑ New



Gwangyang

| Cell Name | | |
|-----------|----------|----------|
| KR3F4H00 | KR647B14 | KR647B26 |
| KR4F4H20 | KR647B15 | KR647B27 |
| KR5F4H21 | KR647B16 | KR647B34 |
| KR5F4H22 | KR647B17 | KR647B36 |
| KR647B06 | KR647B22 | KR647B37 |
| KR647B07 | KR647B23 | KR647B46 |
| KR647B12 | KR647B24 | KR647B47 |
| KR647B13 | KR647B25 | - |



Gunsan

| Cell Name | |
|-----------|----------|
| KR5F2O34 | KR5F4C12 |

↓ Update



Gunsan

| Cell Name | | |
|-----------|----------|----------|
| KR4F2O40 | KR666D94 | KR656B03 |
| KR4F4C20 | KR666D95 | KR656B04 |
| KR5F2O43 | KR656B01 | - |
| KR5F4C21 | KR656B02 | - |



Busan

| Cell Name | |
|-----------|----------|
| KR4G3A40 | KR659C72 |
| KR4G3B30 | KR659C73 |
| KR5G3A44 | KR658D90 |
| KR5G3B33 | KR659C81 |
| KR659C71 | KR659C82 |
| KR659C83 | KR658D00 |
| KR659C91 | KR659C92 |
| KR659C93 | - |



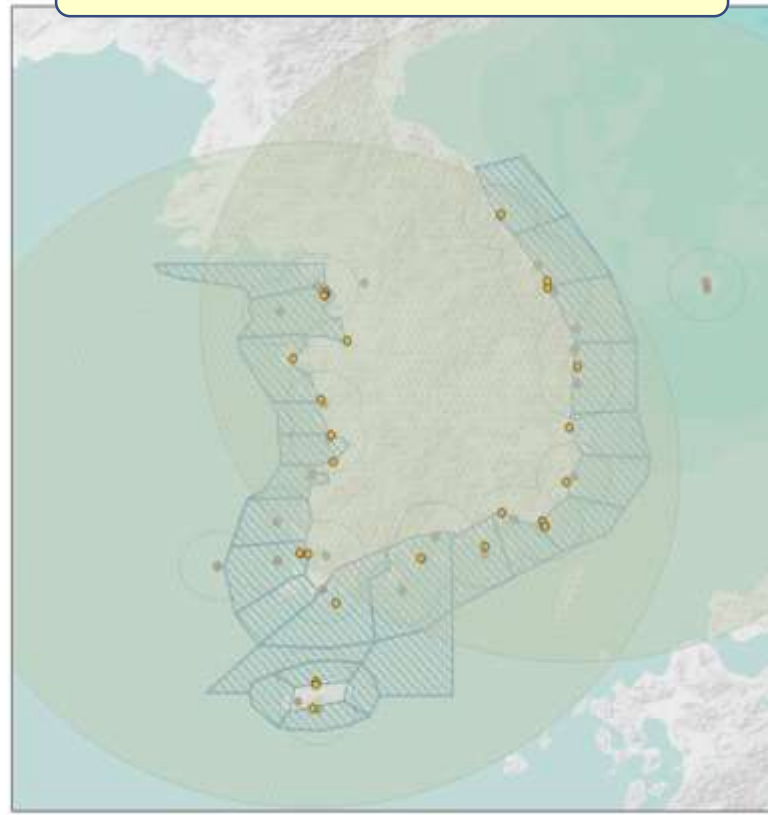
Creation of S-100 test datasets

- Creation of NPUB TDS

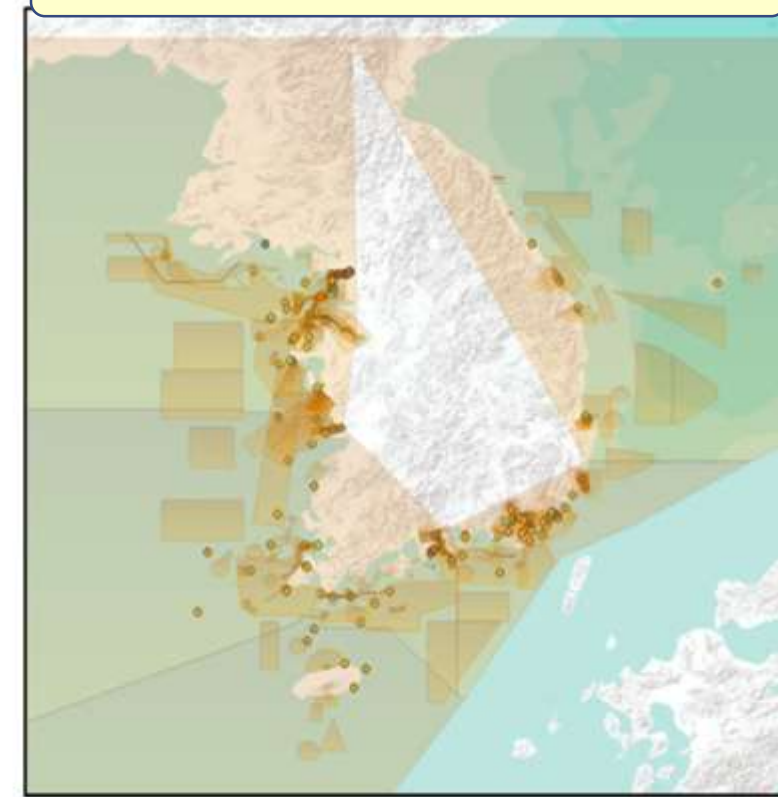
S-122 Marine Protected Area



S-123 Radio Signals

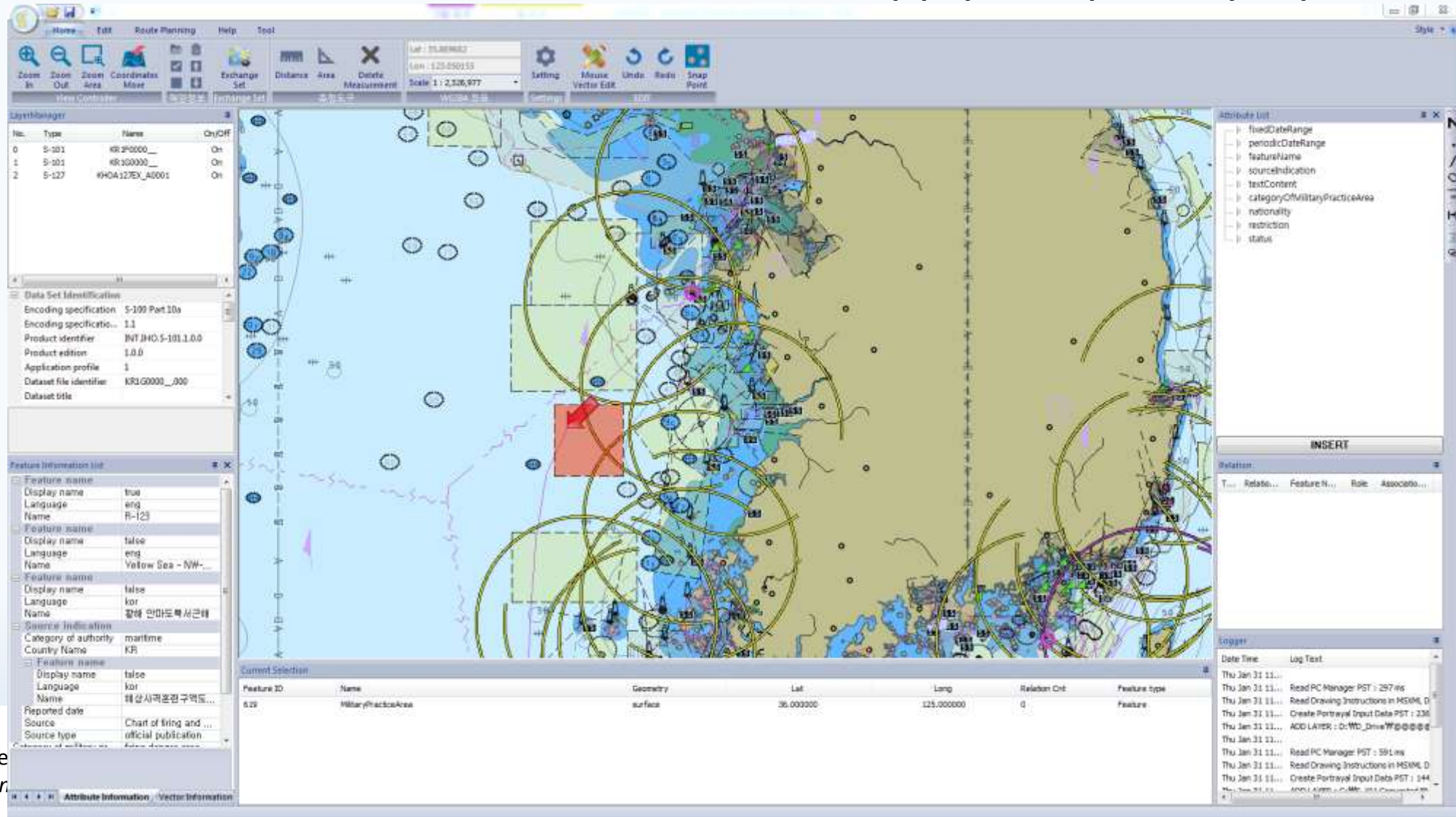


S-127 Marine Traffic Management



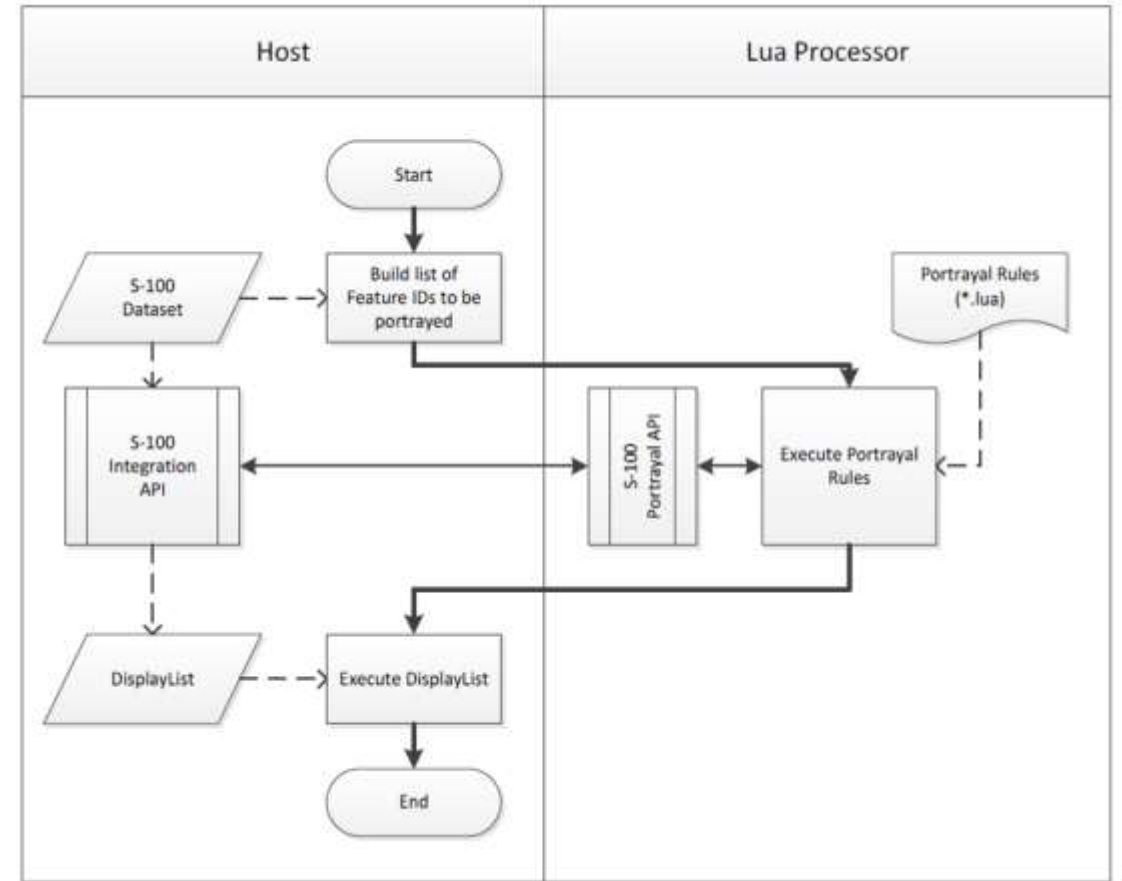
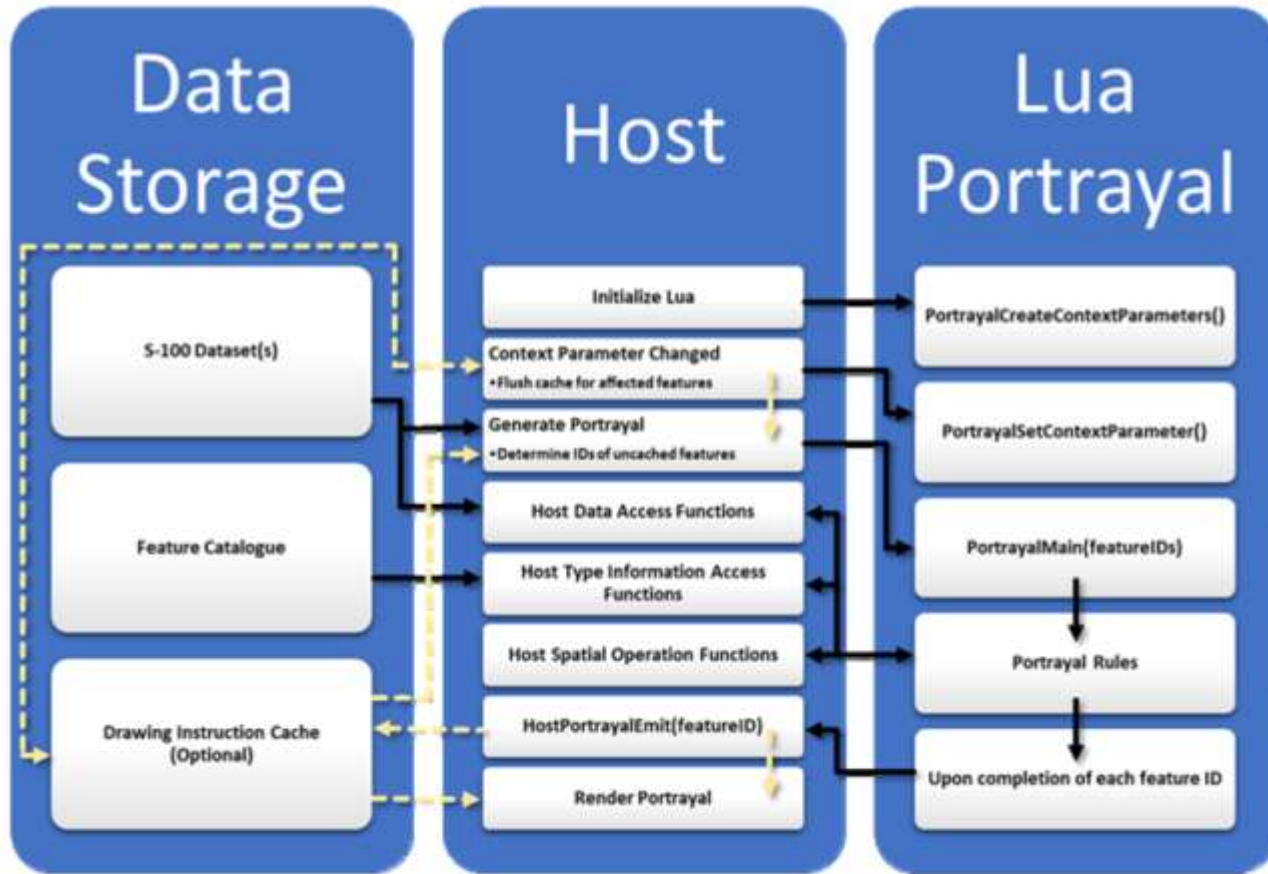
Update of the S-100 Simple Viewer

- Focused to test N PUB test datasets and apply Lua portrayal process



Update of the S-100 Simple Viewer

- Lua process



Update of the S-100 Simple Viewer

- Lua process



Download

[source](#) · [binaries](#) · [previews](#) · [logos](#) · [tools](#) · [test suites](#) · [extras](#) · [license](#) · [versions](#) · [donations](#) · [live demo](#)

❖ Source

Lua is [free software](#) distributed in [source code](#). It may be used for any purpose, including commercial purposes, at absolutely no cost.

All [versions](#) are available for [download](#). The current version is [Lua 5.3](#) and its current release is [Lua 5.3.5](#).



[lua-5.3.5.tar.gz](#)

2018-06-26, 297K

md5: 4f4b4f323fd3514a68e0ab3da8ce3455

sha1: 112eb10ff04d1b4c9898e121d6bdf54a81482447

❖ Building

Lua is implemented in pure ANSI C and compiles unmodified in all platforms that have an ANSI C compiler. Lua also compiles cleanly as C++.

Lua is very easy to build and install. There are [detailed instructions](#) in the package but here is a simple terminal session that downloads the current release of Lua and builds it in Linux:

```
curl -R -O http://www.lua.org/ftp/lua-5.3.5.tar.gz
tar xzf lua-5.3.5.tar.gz
cd lua-5.3.5
make linux test
```

Update of the S-100 Simple Viewer

- S-100 4.0 part 13 – Scripting
 - This Part defines a standard mechanism for including scripting support in S-100 based products. Scripting provides for processing of S-100 based datasets via script files written in the Lua programming language.
- Standard Script Functions (C#)
 - Standard Catalogue Functions
 - Standard Host Functions

Lua <-> C++
Table type (Stack)

13-8.2 Standard Host Functions

C#

```
string[] HostGetFeatureIDs()  
string HostFeatureGetCode(string featureID)  
string[] HostGetInformationTypeIDs()
```



13-8.2 Standard Host Functions

C++

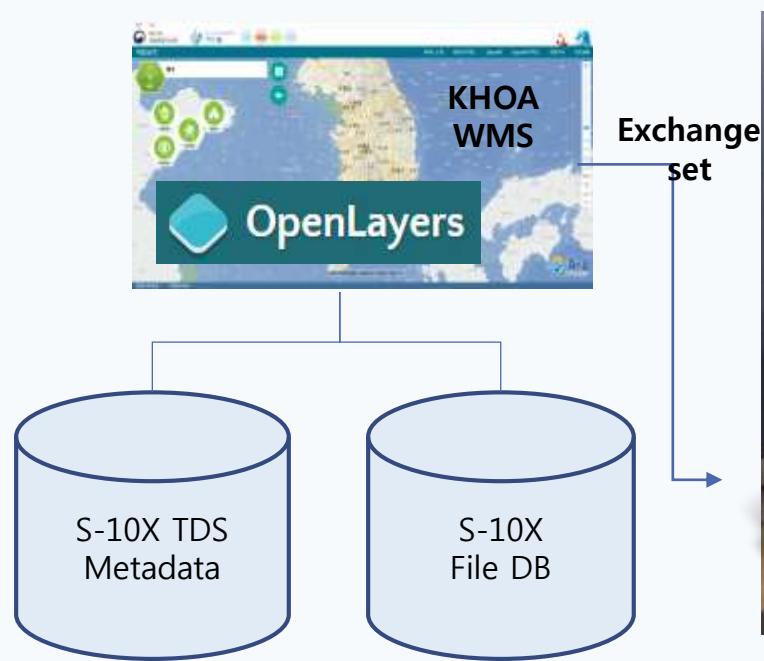
```
int HostGetFeatureIDs(lua_State* L)  
int HostFeatureGetCode(lua_State* L)  
int HostGetInformationTypeIDs(lua_State* L)
```



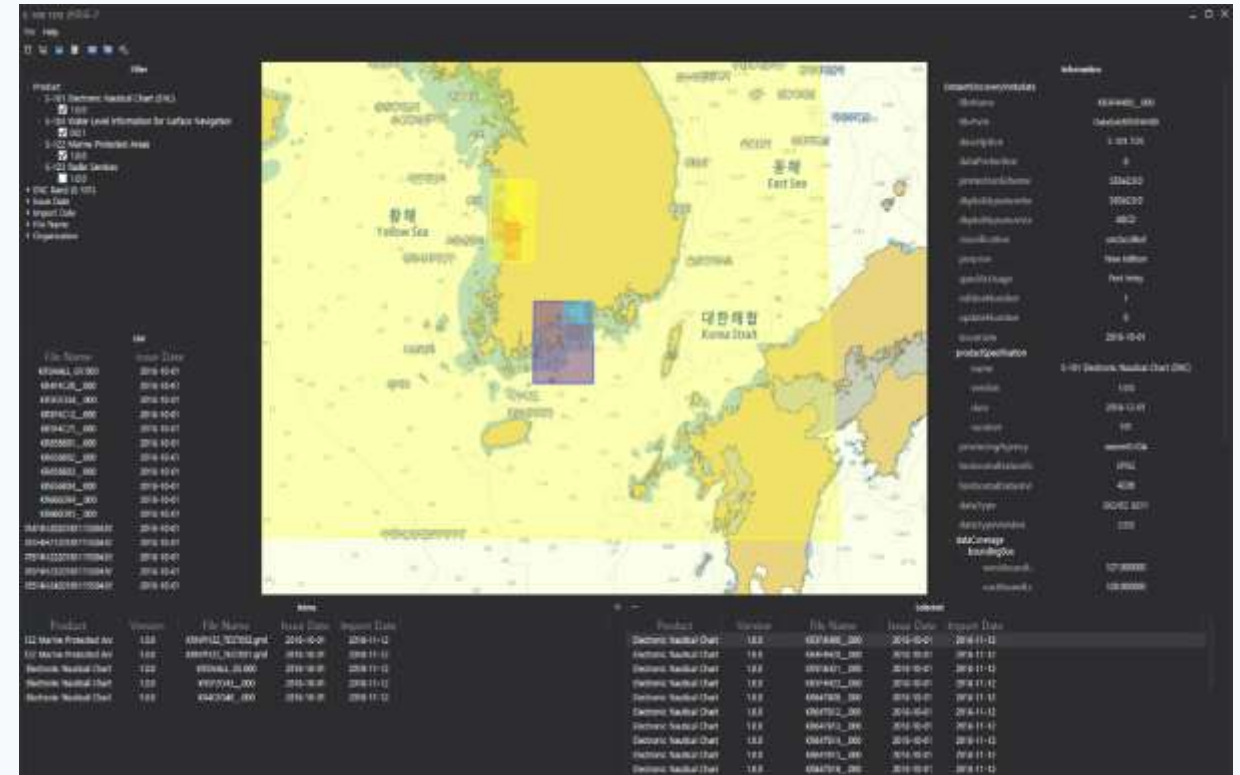
Development of the TDS Management System

- Metadata DB and TDS management system

TDS Management System

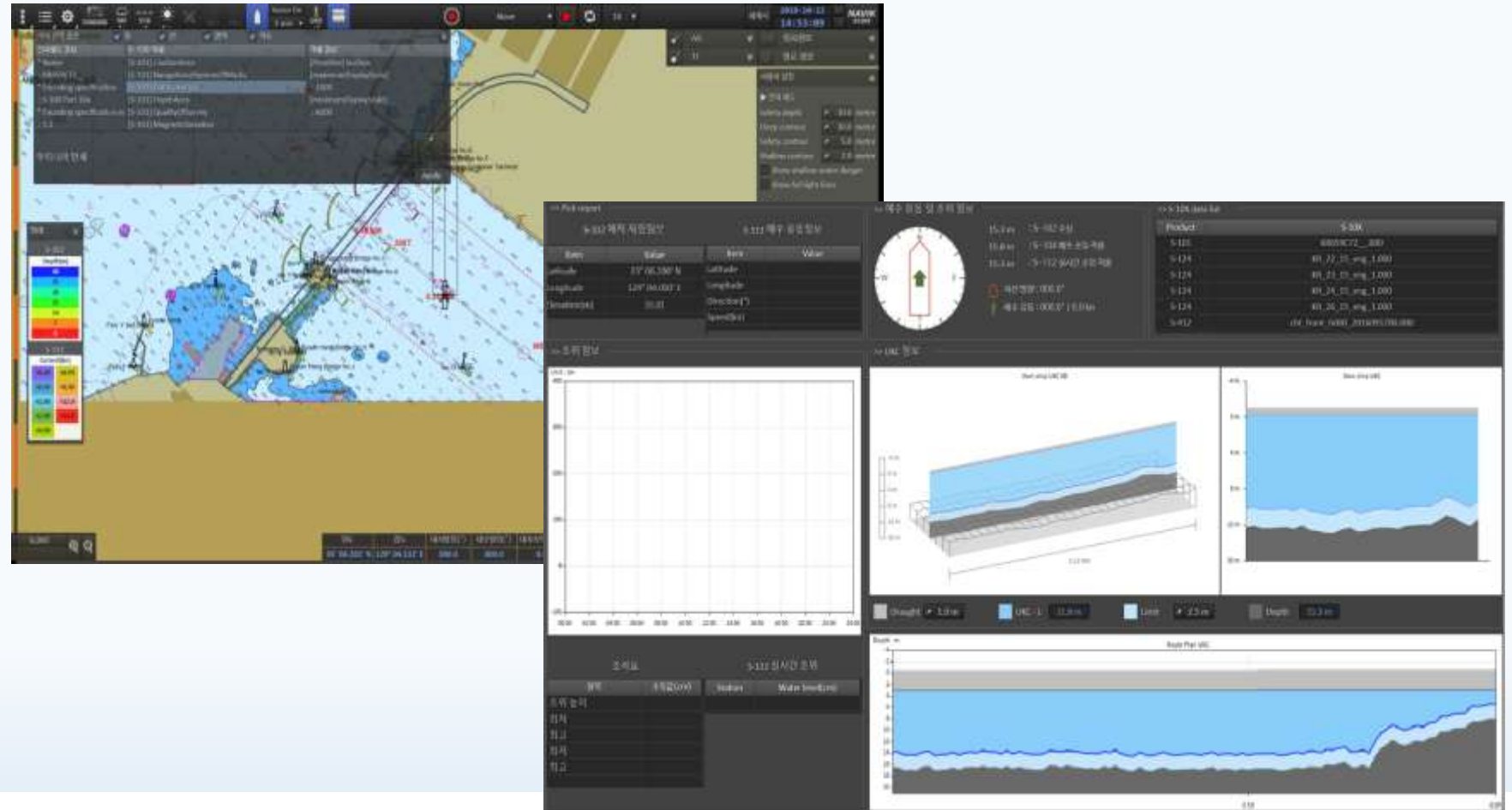


S-100 Test system



S-100 Test System (Shore based ECDIS) (Phase 6)

- S-100 test system for Phase 6



Conclusions and Recommendations

- Conclusions
 - The project focused the development of S-100(Phase 3) Viewer and Shore based ECDIS(Phase 6) according to the S-100 Test Framework.
 - In the reporting period, KHOA tested the NPUB test datasets and tried to apply the Lua portrayal process in the S-100 simple viewer
 - The TDS Management system was developed to manage S-100 data and package S-100 exchange set
 - The ECDIS SW in the S-100 test system was changed as the latest version and the UKC function was improved to support the 2.5D chart.



Conclusions and Recommendations

- Recommendations
 - KHOA is trying to apply the Lua portrayal process in the S-101 simple viewer,
 - but due to lack of experience of the Lua technology and information to develop the Host Function
 - the SW development is very slow and some difficulties are encountered
 - it is recommended to discuss related issues.
 - Provide Standard C++ host function in the Part 13 Scripting
 - Explanation of exchange data contents between Lua and Host function
 - Explanation of data types and structures used in the Lua Rules
 - Provide an example(tutorial) source code of how to use the Lua process

