Research on the Software Development Architecture of Naval Simulation Training Standard Console

Xu Jing, Xu Ming, and Li Tie

***Abstract*—A component based Architecture is proposed to meet the requirements of the software development architecture of naval simulation training standard console. The Standard Element Database (SDD) and management system, the Foundation Classes Library (FCL), the Standard Object Model Development Tool(S-OMDT) are developed which make developing simpler and rapid.**

***Index Terms*—Simulation training, standard console, software development architecture.**

I. Introduction

To make simulation system standard, the Defense Modeling and Simulation Office (DMSO) proposed the High level Architecture (HLA) as the distribute simulation standard [1]. New simulation systems are developed base on HLA. But different systems employing different RTI cannot be connected with each other because the RTI are not compatible, even systems using same RTI are not compatible too, the reason we will represent in the following section.

To make the development of naval simulation training standard console simpler, some open source simulation engines are developed, such as the Combat Simulator Project and Delta3D. The Combat Simulator Project is a flight simulator engine and Delta3D is game engine developed by Naval Postgraduate School (NPS) [2], [3]. These complicated engines supporting 3D scene are not easy to learn and are not very suitable for console simulator mainly composed of instruments. When developing virtual instrument, many programmers employ GLStudio, an useful software with friendly and convenient tool. However, it is not free and gives nothing except virtual panel.

This paper focuses on developing naval simulation training standard console, analyses the characteristics and requirements of console, and proposes a solution based on component technology, which is named as Standard Console Simulator Development Architecture (SCSDA).The design of SCSDA is as followings.

II. Foundation Class Library (FCL)

We can list the general problems programmers encounter when building console simulators, scene of instrument panels, GUI, modeling, database manager, network, file IO, serial port communication, sound, memory, threads synchronization. To meet these problems, SCSDA develops a Foundation Classes library (FCL), some classes of which are from open source software, such as CEGUI, OpenAL, Xerces *et al.*, See Fig. 1.

A. CBasePart

The CBasePart class is responsible for simulating visible instruments on consoles, which include switch, knob, indicator, led, panel, keyboard and screen. These object-oriented classes are easy to use and respond well to messages from outside. With these classes, programmers can build the virtual parts effectively in a short time. When necessary, programmers can also custom self-defined part class derived from CBasepart or other part classes to simulate special instruments.

B. CBaseGraph

For rendering, SCSDA designs the CBaseGraph class. It derives some classes, CBaseGDI, CBaseGUI, CBaseGL and CBaseDX, adopting different technology, such as GDI, GUI, OpenGL and Direct3D. They are used as individual's will.

C. CBaseAudio

SCSDA's audio is handled through the CBaseAudio class which consists of two classes, one based on the Open Audio Library (OpenAL) and one based on DirectSound. CBaseAudio consists of a number of functions that allow a programmer to specify the objects and operations in producing high-quality audio output, specifically multichannel output of 3-D arrangements of sound sources around a listener. It can handle sound source directivity and distance-related attenuation and Doppler effects, as well as environmental effects such as reflection, obstruction, transmission, reverberation.

D. CBasePart

SCSDA uses the CBaseDataManager class to manage data in simulator system. Three derived classes from it are CBaseINI, CBaseXML and CBaseComm. CBaseINI handles .ini files or binary files with self-defined format. The XML files are handled through CBaseXML which is derived from Xerces. CBaseComm, which performs data communication, derives three classes, CBaseSocket, CBaseCom and CBaseShareMem. Developers can use them to handle TCP/IP programming, serial communication and memory sharing.

E. CBaseDB

CBaseDB performs functions of database management, including connection, query, modifying, appending and etc.

Manuscript received December 10, 2012; revised February 20, 2013.

The authors are with the Dalian Naval Academy/Simulation Training Center, Dalian, China (e-mail: xujing2010@163.com, xioneering@yahoo.com.cn, 13555965759@139.com).

DOI: 10.7763/IJMO.2013.V3.253

F. CBaseModel

There are many models employed in military console, such as filtering models, firing models, movement models, flow controlling models and other special models. SCSDA uses CBaseModel to manage them.

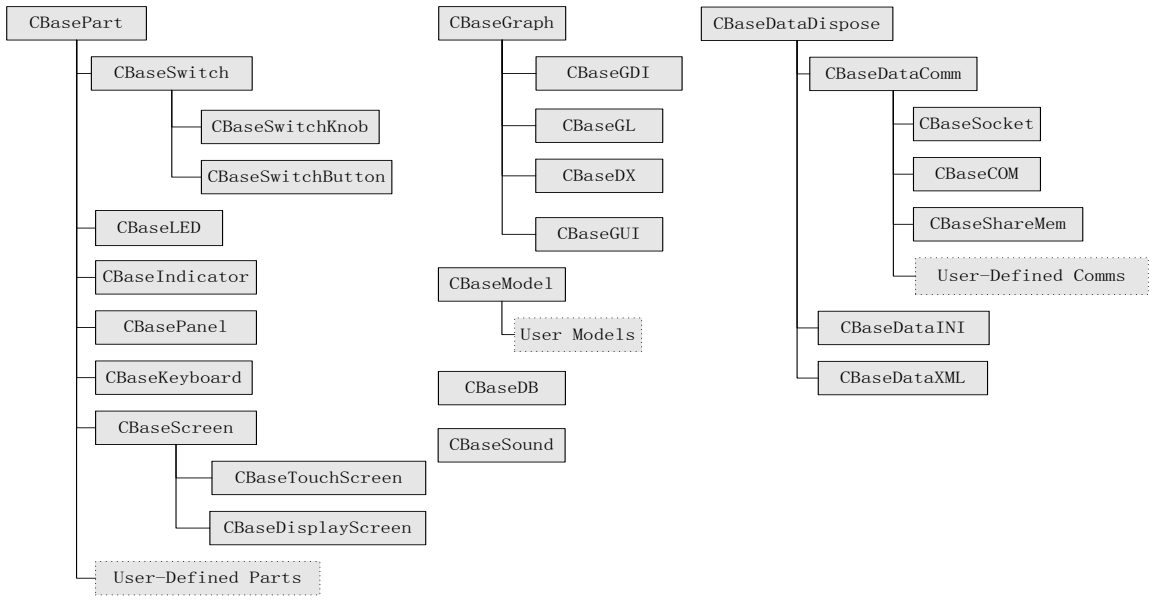

Fig. 1. Foundation classes.

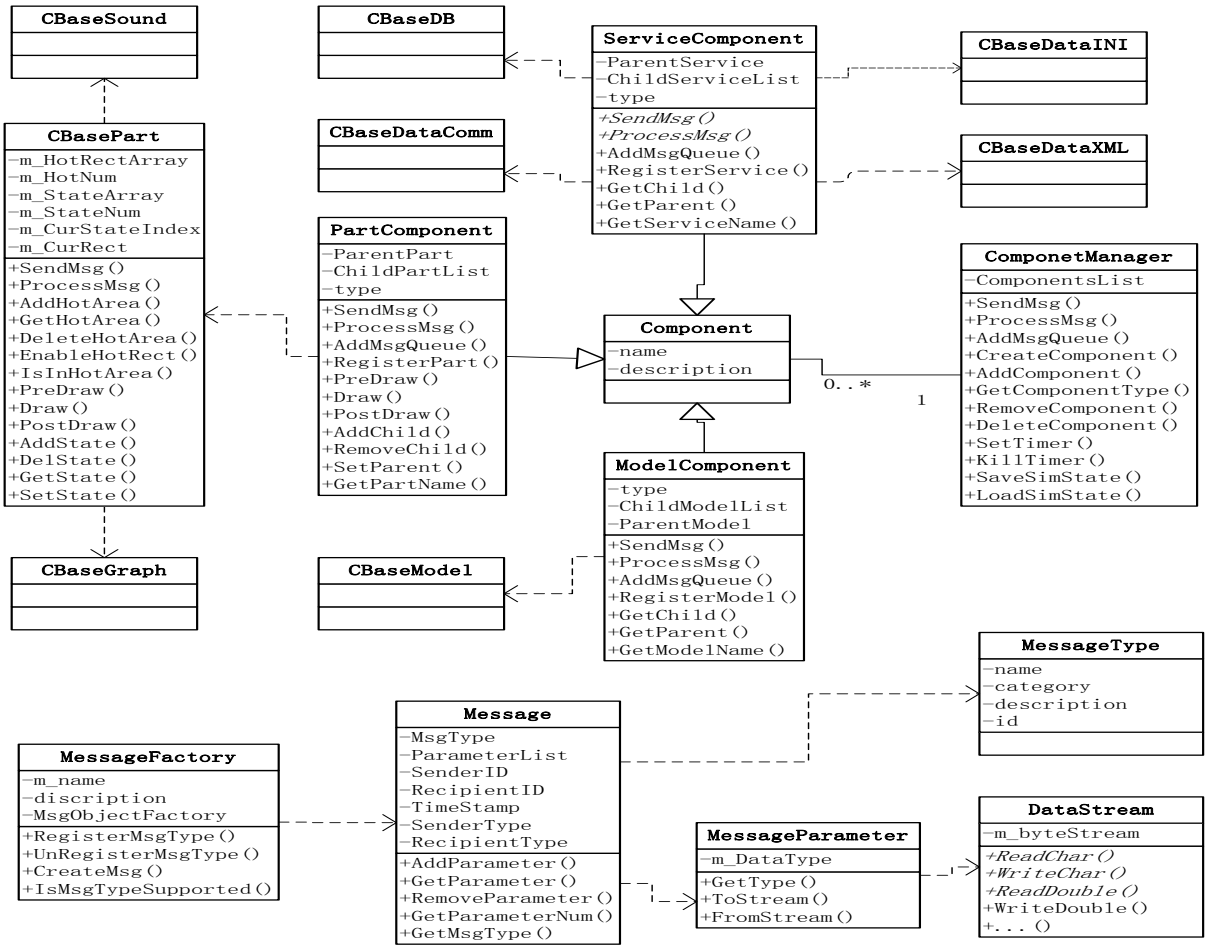

Fig. 2. SCSDA classes diagram.

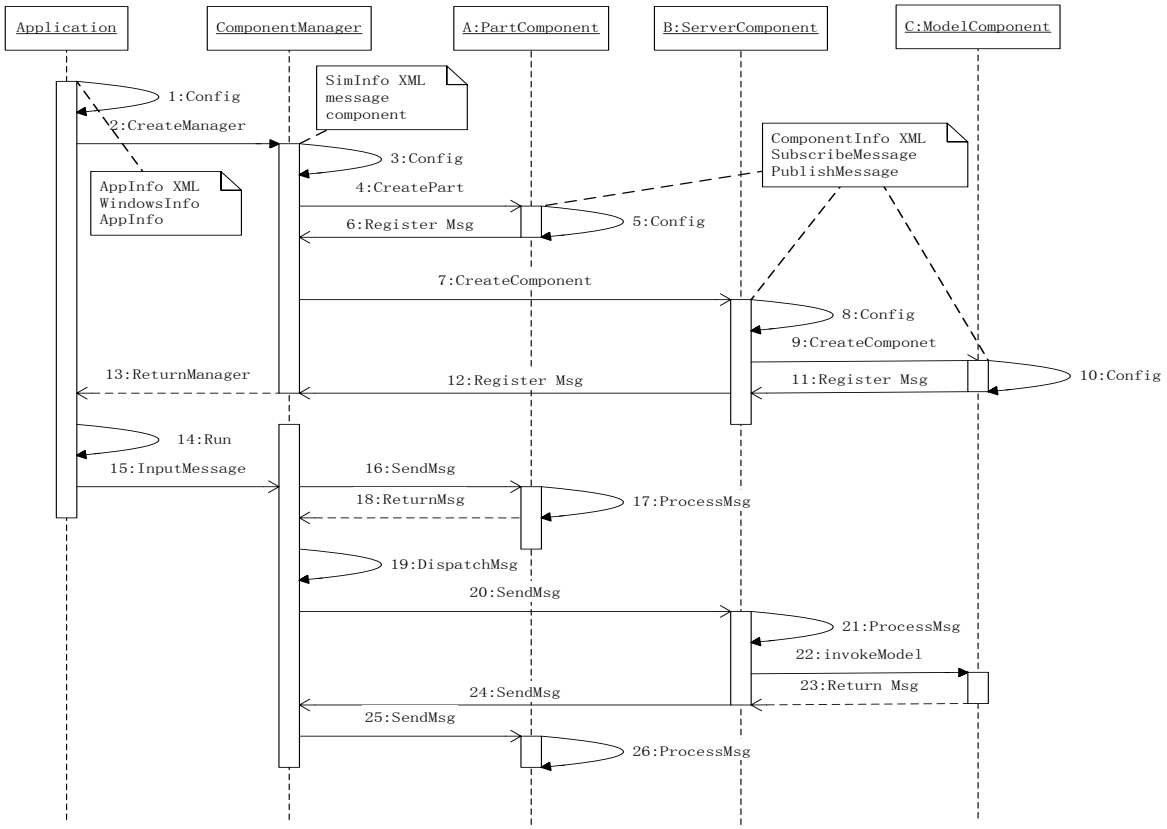

Fig. 3. Example of application sequence diagram.

III. Component Manager and Components

Systems developed by SCSDA run as components and component manager. Components consist of part, service and model components. The three types of component are responsible for different simulator functions; the Part Components for visible instrument panels; the Model Components for Modeling; the Service Components for other services. The Component Manager is the core of the systems developed by SCSDA; the glue that holds everything in the systems together. It is responsible for managing part components, service components and model components, ensuring messaging and inter-object communication, and directing component behavior. It knows which components exist, what parts are interested in which messages, and what models exist in the simulation. Moreover, it is responsible for making sure messages get to the interested components as well as handling low-level events from the simulator system such as frame and pre-frame events. The Fig. 2 class diagram shows the architecture of both Component Manager and components. The Fig. 3 sequence diagram illustrates the processes between the Component Manager, its part components, service components and model components.

IV. Simulator System Xml Files

SCSDA employs many config files when working. The config files are flexible XML files shown a few in Fig. 3, and here are more descriptions of the XML files. There are four main types of XML files:

1) Application XML file, which is loaded when application initializing, is responsible for recording windows number, position, size, information of component manager, *et al*.
2) Simulation system information XML file. The file resembles FED file of HLA. It describes all information of messages and components. The Component Manager employs it to handle messages and direct components behavior. The XML file should be defined before building a simulator system, and every component uses it to define component information XML file, which will be described next.
3) Component information XML file is responsible for information of subscribing and publishing messages, parent and child component, as well as additional XML files such as CEGUI XML files and parameter XML files.
4) Message-FOM mapping XML file is a special file used in HLA Components. It performs task of converting local messages to HLA traffic. Therefore, the HLA component can connect local simulator with remote ones.

V. Standard Data Database (SDD)

Standard Data Database (SDD) is the core of SCSDA architecture. It is impossible to develop standard simulators without SDD. Data in SDD is defined according to documents of military training simulator standards which stipulate for all data in military simulators.

When developing some special simulators, developers may not be able to find what they need in SDD. Some Reserved Data (RD) is pre-defined in SDD consequently.

When SDD could not meet the requirement, developers use these RD temporarily; of course, they need report these data to the special institute. Therefore, the SDD will be updated periodically.

VI. SDD Management System (SMS)

STADA uses SDD Management System (SMS) to manage the SDD. It is an independently subsystem of SCSDA, responds for maintaining SDD, including appending, enquiring, modifying, deleting, reporting, data importing and exporting. On the other hand, SMS submits application programming interfaces (API) of SDD.

VII. Standard Object Model Development Tool (S-OMDT)

SCSDA develops the Standard Object Model Development Tool (S-OMDT) [4]- [6]. The standard data in SDD can be imported into S-OMDT which can generate standard FED file or XML file of FOM/SOM. Therefore, members of federation, even of different federations, are compatible with each other.

VIII. Standard Component Wizard (SC-WIZARD)

Standard Component Wizard (SC-Wizard) is a tool by which developers generate initial code of components. It can generate standard C++ object-oriented classes which expose interfaces shown in Fig. 2.

When developing a special component, HLA service component, we employ the third party tool, FedWizard [6], to generate HLA code framework based on FOM/SOM created by S-OMDT. The HLA component connects simulator with RTI software by message-FOM mapping XML file.

IX. SCSDA's Architecture and Flow

In a word, The SDD and SMS are the core of SCSDA and the base of S-OMDT. When developing, the SC-Wizard creates components with standard interfaces at first; then programmers develop components employing FCL. The Fig. 4 illustrates a high level relationship between SDD, SMS, S-OMDT, SC-Wizard and FCL.

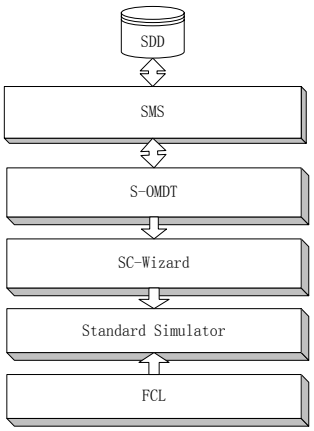

Fig. 4. High level view of SCSDA.

X. Conclusion

This paper gives a description of solution of developing standard simulator which is named SCSDA. It focuses on standardization, compatibility and component of simulator; develops some foundation classes to simplify programming; builds standard data database and toolset to help system compatible; employ component to make process of development flexible. Therefore, SCSDA offers an outstanding choice for naval simulation training standard console developers. It could be used for other purposes when new classes are adopted in FCL if desired.

References

[1] S. White, "Using an HLA Maritime Combat System Simulation Environment to Investigate Force and Coalition Interoperability," in *Proc. 2002 European Simulation Interoperability Workshop*, Middlesex, UK, 2002.

[2] J. M. Nolan, "Games for training: Leveraging commercial off the shelf multiplayer gaming software for infantry squad collective training," Naval Postgraduate School, 2005.

[3] P. Mcdowell, R. Darken, and E. J. Sullivan, "Delta3D: a complete open source game and simulation engine for building military training systems," *JDMS*. 2006, vol. 3, no. 3, pp. 143-154.

[4] J. X. Gong, W. Zhong, J. Huang, X. G. Qiu, and K. D. Huang, "Application of base object model in HLA's simulations," *Journal of System Simulation*, vol. 18, suppl. 2, pp. 327-331, 2006.

[5] Q. Zhang, Q. J. Yin, and K. D. Huang, "A conceptual study of base object model (BOM)," *Journal of System Simulation*, vol. 17, no. 7, pp. 1667-1670, 2005.

[6] J. Yin, J. G. Hao, and K. D. Huang, "Research and implementation of automatically generating federate software framework from high level architecture object model," *Journal of System Simulation*, vol. 19, no. 6, pp. 718-722, 2002,

Xu Jing is a professor, he was born in Wuxi, Jiangsu Province, China on March 15, 1969. He received the Ph.D. degrees in computer application from Dalian University of Technology in 2002. Since 1994 he was employed by the Dalian Naval Academy, and he has worked in the areas of modeling and simulation of naval operation during his years in Dalian Naval Academy, he was appointed head of the simulation training center in 2006. He has published over 50 articles in his research areas.

Xu Ming is a lecturer, he was born in Wuxi, Jiangsu Province, China on June 12, 1980. He received the M.A. degrees in computer engineering Dalian University of Technology in 2011. Since 2008 he was employed by the Dalian Naval Academy, and he has worked in the areas of modeling and simulation of naval operation during his years in Dalian Naval Academy. He has published over 10 articles in his research areas.

Li Tie is a lecturer, he was born in Miro, Hunan province, China on Dec. 5, 1974. He received the Ph.D. degrees in computer engineering from Northwestern Polytechnical University in 2012. Since 2001 he was employed by the Dalian Naval Academy, and he has worked in the areas of modeling and simulation of naval operation during his years in Dalian Naval Academy. He has published over 30 articles in his research areas.