Research Article
Building the System Interface Management Environment for the Development of Complex System

[1]Sung Gyun Oh and [2]Peom Park
[1]Department of Systems Engineering,
[2]Department of System Engineering Industrial Engineering, Ajou University, Suwon, Korea

Abstract: Advanced systems have common characteristics of complexity as the level of their demanded emergent capability and the resulting interfaces among their components increase. These characteristics make it difficult to manage the interfaces and the failure of the management can lead to the failure of development projects. This study proposes a model-based systems engineering approach to facilitate the interface management for an IPT environment. A demonstration of the proposed approach to the magnetic levitation railway development project is provided to identify and control interfaces.

Keywords: DSM, interface management, MBSE, system of system

INTRODUCTION

The state-of-art systems are evolving towards larger size and more complex interaction among sub-systems. In other words, the management of interface has become difficult as the target system has an increasing number of internal or external interfaces. Management of system interface is very important for the development of system architecture. That is because inappropriate management of internal/external interface in system may lead to the failure or delay of system development (Lane, 2009; Han *et al*., 2014). In particular, the System of System operated through interaction of multiple systems involves participation of many project contractors for each system. These organizations have unique interface development environments which are difficult and consequently may cause conflict among organizations.

This study deals with the cases of interface management environment for the project preceding the magnetic levitation train project.

Problem statement: First of all, In this case, we can identify the problems mentioned earlier. Some organizations of companies and many research institutes which are participating in this project do not have clear in-house regulations pertaining to the interface management. In addition, the remaining companies and organizations that have enforced such regulations have different management environment. Therefore, it was necessary to create the interface management environment that will be used jointly by those organizations to help ensure success of the project.

Second, this project has some problem with the development process. As the development schedule did not take the design process into consideration, the interface control has become extremely difficult. Figure 1 shows the project program of each development organization identified in RFP of this project. While the system engineering contractor A.A generates the operational concept, contractor B.A undertaking the development of train system, one of the system components, determines the Specification.

Contractor C.B who develops the railway simultaneously carries out the detail design of this system. In that way, different processes result in extreme difficulty with interface and shape control for each contractor's outcomes and consequently, may lead to the re-design or fatal failure of system.

The purpose of this study was to provide a development environment for the Integrated Product Team (IPT) of each organization to carry out common interface management activities for the preceding research on magnetic levitation train project.

METHODOLOGY

System engineering process: In general, the development precedents can be found for commercial system development projects. The risk and operational burden can be mitigated significantly if the legacy architecture is reused.

Sage defined the product re-engineering as reconfiguration of conventional products' internal mechanism or functions into functional and non-functional form through the inspection, research, identification and modification of such internal
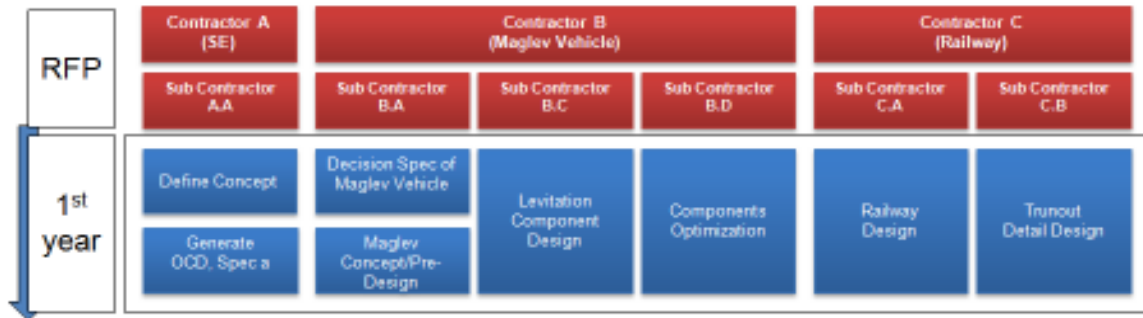
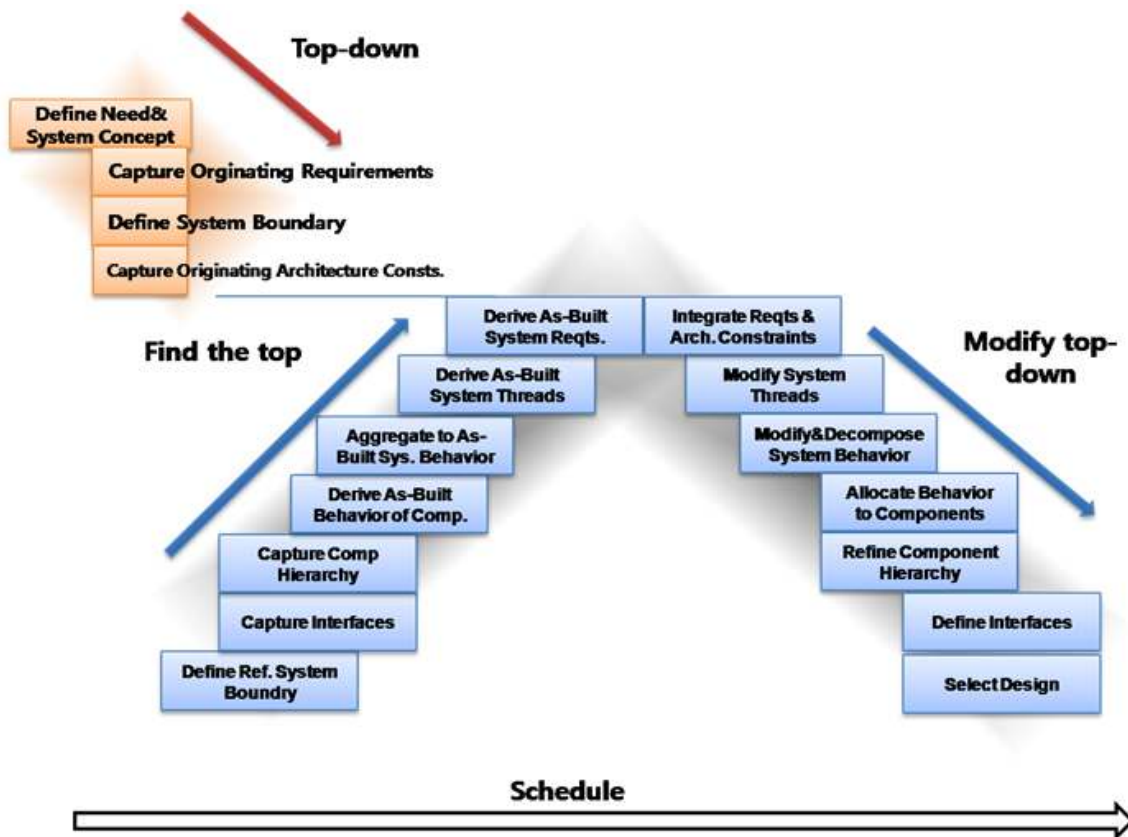Fig. 1: Project program of maglev system

Fig. 2: System re-engineering process

mechanism or functions in order to apply new technology while achieving essential objectives of conventional products (Sage and Lynch, 1998; Sage and Rouse, 2009). According to the definition drawn by Sage, re-engineering means the product process that involves the sequence of forward-engineering, reverse-engineering and re-engineering. Figure 2 presents a diagram of the re-engineering process. The re-engineering process implements the net-engineering process that carries out the requirement definition process by identifying the needs and system concepts of initial stakeholders. In addition, the re-engineering process carries out reverse-engineering process that derives system requirements by analyzing the reference system. Finally, along with the requirements that have been derived in that way, the re-engineering process is implemented to carry out the solution definition process again by integrating the requirements that have been generated earlier.

While performing the re-engineering process, the interface obtains interface relationship between the system of reference system or sub-components in the process of reverse-engineering and is identified through the solutions definition process for the target system in the re-engineering process. The interface, which has been identified by implementing the re-engineering process in the aforesaid way, is managed through CASysE Tool.
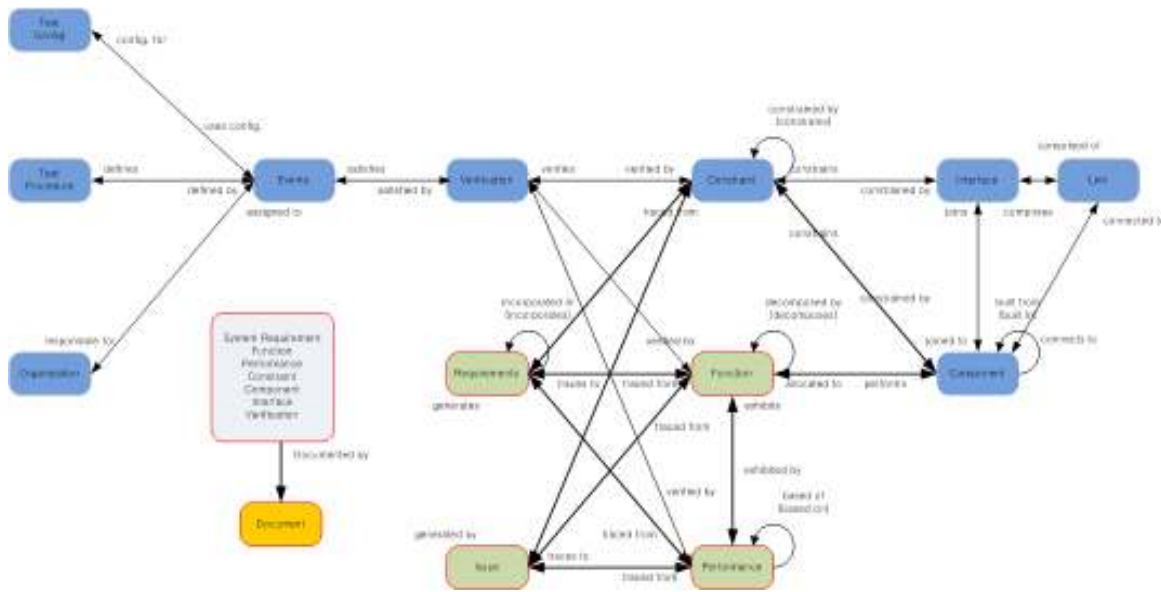
Fig. 3: The schema for interface control

CASysE tool: Currently, many system engineers are using the 'Computer Added System Engineering Tool' to perform the Model-Based System Engineering design and many related cases can be found (Do and Cook, 2012; Bonanne, 2014; Matar *et al*., 2014; Scheeren and Pereira, 2014; Góngora *et al*., 2015). MBSE presents reasonable alternative systems through the modeling and simulation from various viewpoints at system level and the response to the vast amount of information by using the computational support tool, thereby reducing the number of prototypes that are created. Moreover, MBSE can ensure traceability for various requirements ranging from stakeholders' requirements to system validation requirements and increase productivity of works. The system design data, built into database, can facilitate the use and reuse of information related to the existing system in the development of similar systems

In this regard, the use of system engineering computational support tool (CASysE Tool) will facilitate system engineering process implementation ranging from the analysis of requirements to the realization of system architecture. Using this tool, the magnetic levitation train project provided simultaneous development environment of many developers and particularly, interface management environment. In this study, the Cradle of 3SL Co was used as the tool.

Then, to build the integrated interface management environment that uses the computational support tool, the definition of schema is required which defines the data elements, relationship among such data elements and attributes of the elements.

Figure 3 presents the schema built for the identification and control of interface in this study. The following Chapter describes the re-engineering process that has been explained above and the interface

identification and control activities using the computational support tool that supports the interface management environment within the re-engineering process.

Case study:

Magnetic levitation (maglev) transit system: In this Study, we discuss the cases of magnetic levitation (maglev) transit system project. Magnetic levitation train system makes the vehicle levitate at some height over the track and has the system operation concept different from that of existing railway system (Han *et al*., 2014). Rail system has high maturity architecture and therefore the use of legacy architecture can be used when developing new systems. This case involved application of re-engineering process to reduce development effort and cost and used the light train system as legacy system. The re-engineering process will be introduced in the following section.

Figure 4 presents the diagram of physical architecture of magnetic levitation train system that reuses the architecture of conventional train system. As the vehicle is changed to magnetic levitation train, new requirements have arisen for the form item of other 6 systems that interact with this subsystem. The functions of magnetic levitation train system are realized through complex interactions between subsystems. Therefore, strict and rigorous management is required to resolve the interface problems arising from the change in the form items of each subsystem.

Selection of legacy system: The system that has the most similar operation concept is selected based on the needs and concept which have been defined from customers and stakeholders. Urban magnetic levitation train system has the operation concept which is the most similar to that of Lightweight Train (LRT) system.
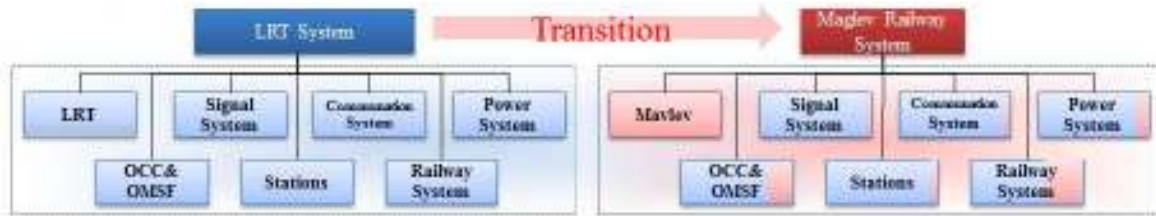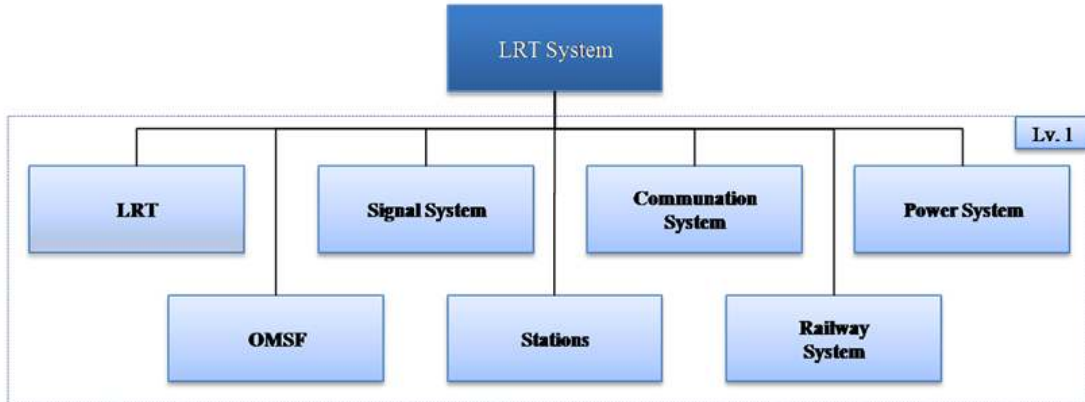
Fig. 4: Reusing architecture of legacy system

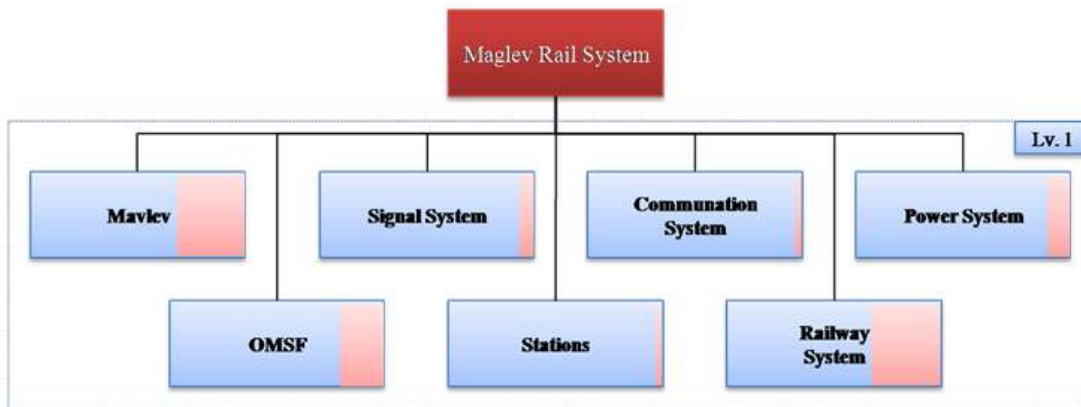Fig. 5: Physical hierarchy of LRT system

Fig. 6: Physical hierarchy of maglev system

The selected system, which is the legacy system, can sequentially derive the physical-functional requirement architecture through the reverse-engineering process of re-engineering process.

Identification of LRT system interface: The interface and physical hierarchical structure of system were verified through the related documents such as OCD, System Spec., URD, ICD, IDD, etc., of the lightweight train system which was selected as legacy system. LRT consists of 5 subsystems as shown in the Fig. 5.

As the physical hierarchical structure of these systems is clearly identified, the interface between them becomes clear. The identified interfaces are maintained

in the new system, excluding the interfaces that are not affected by the physical components that will be changed or added in the re-engineering process and the interface that have become unnecessary due to such physical components. If the system requirements are finally identified after the functions are derived through physical hierarchical structure, the re-engineering process is implemented by integration with original requirements that have been obtained through net-engineering process.

Figure 6 presents a diagram of physical hierarchical structure refined through the re-engineering process. Although the change cannot be figured out at the Lv.1 level, we can ascertain the functional/physical
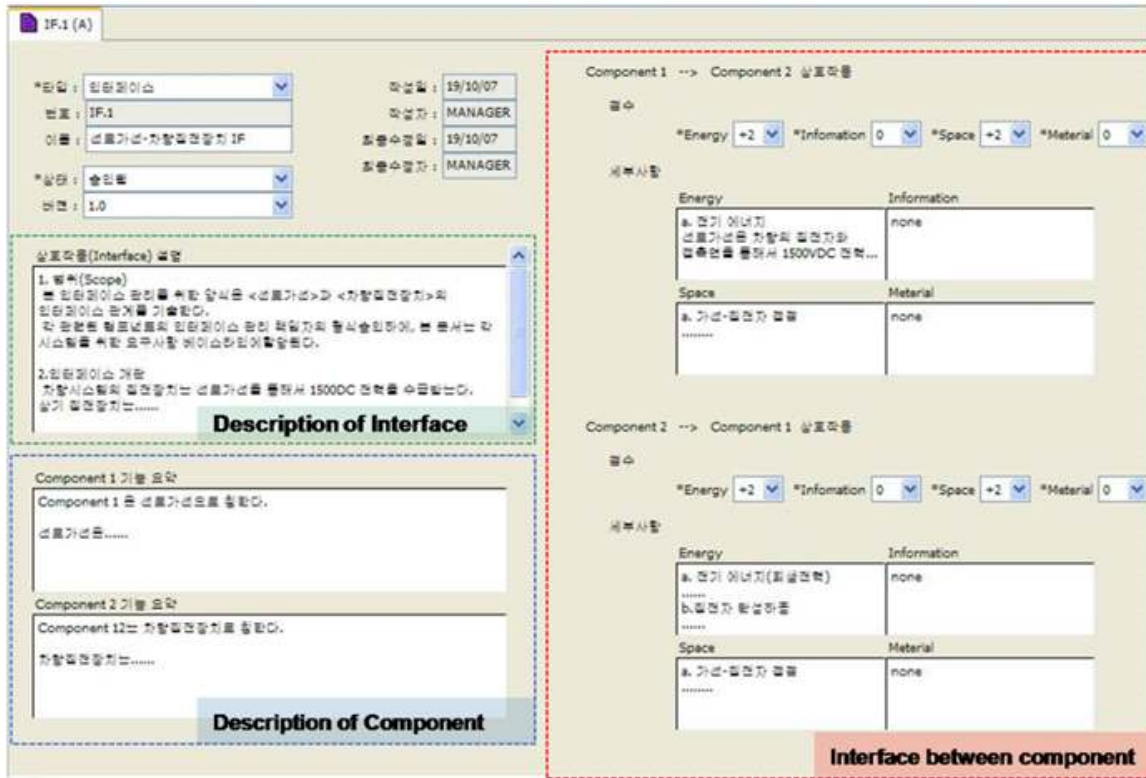
Fig. 7: Interface form

hierarchical structures that have been changed due to the integrated requirements as the level becomes lower.

The area marked in red within each system block represents the change in physical components as compared to the reference system. The width of the area is proportional to the quantity of modified components. In addition, the width of the area is proportional to the quantity of internal/external interface of the system affected (which therefore requires the review by ICWG) by the change in components.

The interfaces that have been identified through aforesaid process are fed into the interface form presented in Fig. 7. This interface form includes the detailed description that specifies the name, scope and function of interfaces, along with the explanation of components targeted for interface and interface relationship among components.

In this study, we applied DSM to ensure easy representation of interfaces. It was found that this model was applied extensively for development as this model allows the interfaces between system components to be easily represented and understood. This study specifies the 4 types of definitions for interface category between components to apply DSM, as presented by Pimmler and Eppinger (1994). Scores were added, depending on the importance of energy exchange, information exchange, materials exchange, physical space and arrangement and interface among components.

The clustering which can be implemented through DSM provides powerful system integration analysis function (Pimmler and Eppinger, 1994; Smith and Eppinger, 1997; Rushton and Zakarian, 2000; Browning, 2001; Gorbea *et al.*, 2008; Shamsuzzoha *et al.*, 2010). However, this clustering is not dealt with in this study and was limited to the provision of input form enabling the input of the interface identified through template.

Figure 8 presents the image of input data retrieved through Query View. The input/output relationship between components can be verified through the following Query View.

Interface control: The ICWG is organized when the interfaces obtained from reverse-engineering process and re-engineering process are incorporated into database. The purpose of ICWG is to resolved interface problems which are difficult to be resolved through simple engineer-to-engineer relationship. Figure 9 illustrates the interface management process. ICWG reviews and revises the identified interface requirements to enhance maturity. Moreover, ICWG reviews the effect of interface which may arise from the change in components and resolve problems.

In this study, we applied ICD preparation guideline specified in the Systems Development Life Cycle Guidance Document of the U.S. Department of Justice to provide the view for supporting the interface control

Fig. 8: Query view for interface control
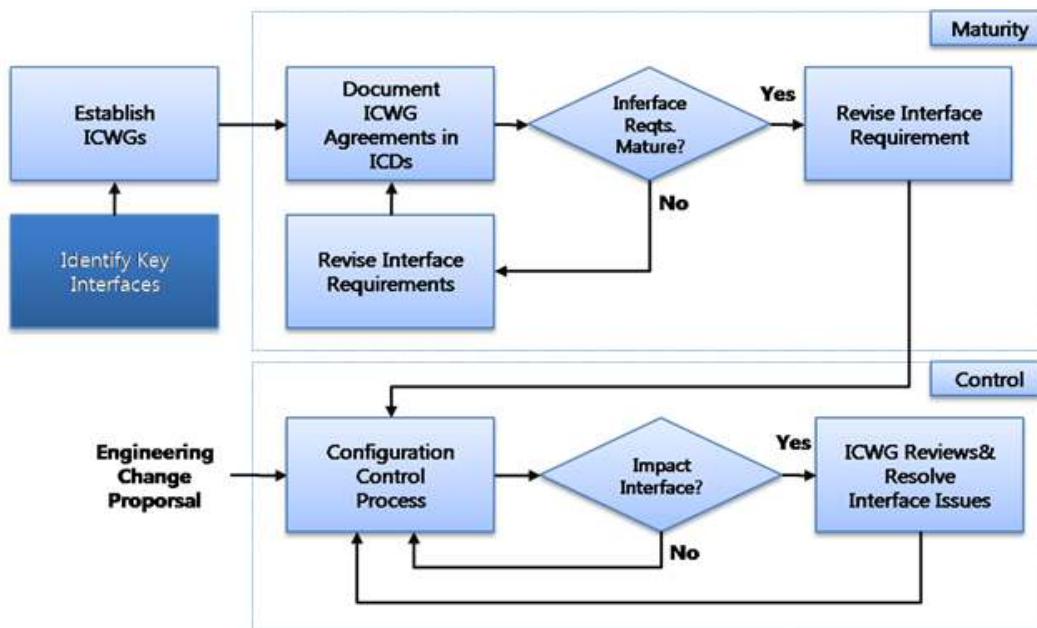
Fig. 9: Interface management process

Table 1: DoJ ICD

No.	Title
1	Scope
2	Concept of operations
3	Detailed interface requirements
4	Qualification methods
5	Notes
6	Appendix
7	Approvals
8	Records of changes

activities of ICWG. The ICD structure presented in this document is shown in Table 1.

The ICD presented in this document defines the scope of system (or components) that has the interface and describes the concerned operation scenario. Besides, the ICD can identify the interface relationship of the two system (or components) through operation scenario to derive interface requirements and specify the requirements for validation. In this study, the tracking relationship between elements described above is shown through the query view below. Figure 10 shows the query view for interface requirement In addition, the ICD presented in this document specifies the revision history.

In this study, we can ascertain the issues presented in the related ICD through the issues as shown in Fig. 11.

DISCUSSION AND CONCLUSION

Recently, many success cases can be found which applied CAsysE to develop effective railway system (Lee *et al*., 2002; Han *et al*., 2012; Bouch and Roberts,

Fig. 10: Query view for interface trace

Fig. 11: Query view for interface issue

2013; Favaro and Sartori, 2013; Prokhorova and Troubitsyna, 2013; Ginisty *et al*., 2014). And regarding the cases of application accommodating specific perspective, there is some case in which CAsysE was applied for analysis of safety (Soubiran and Belmonte, 2012; Prokhorova and Troubitsyna, 2013), ergonomics (Gopinathan *et al*., 2012; Tadros, 2013), sustainability (Azevedo *et al*., 2009; Azevedo, 2010). In this study, we applied CAsysE to carry out Interface Management successfully within the Reverse Engineering Process.

The complex system and magnetic levitation train development project should be pushed forward simultaneously with the development of railway system, communication system and other systems that support the magnetic levitation train chassis, as well as the development of magnetic levitation train chassis. Normal operation of the system can be assured only when no problem occurs in the interface between these systems. Thus, the integrated development team that develops each system should carry out interface management through cooperation with other organizations that develop interactive systems. The problem lies in the fact that it is extremely difficult to maintain undisrupted cooperative relationship when each organization of those teams has different interface management environment or does not have any interface management environment at all. Therefore, all organizations involved in the project needed to use common interface management environment. In addition, it was necessary to support simultaneous development activities of the integrated development teams that carry out development in this organization and many organizations. We presented the measures for identification and control of system interface by using the computational support tool in order to achieve the objectives described above. More specifically, we developed the procedures for identification and control of interface, schema and the view for the implementation of tasks.

ACKNOWLEDGMENT

REFERENCES

Azevedo, K.K., 2010. Modeling sustainability in complex urban transportation systems. M.S. Thesis, Georgia Institute of Technology.

Azevedo, K., B. Bras, S. Doshi and T. Guldberg, 2009. Modeling sustainability of complex systems: A multi-scale framework using SysML. Proceeding of the ASME 2009 International Design Engineering Technical Conference and Computers and Information in Engineering Conference, pp: 1437-1448.

Bonanne, K.H., 2014. A model-based approach to system-of-systems engineering via the systems modeling language. Thesis, Purdue University.

Bouch, C. and C. Roberts, 2013. Developing system models to help Great Britain's railways embrace innovative technologies with confidence. P. I. Mech. Eng. F-J. Rai., 227(6): 677-684.

Browning, T.R., 2001. Applying the design structure matrix to system decomposition and integration problems: A review and new directions. IEEE T. Eng. Manage., 48(3): 292-306.

Do, Q. and S. Cook, 2012. An MBSE case study and research challenges. Int. Council Syst. Eng., 22(1): 1531-1543.

Favaro, J. and G. Sartori, 2013. Introducing model based software design in railways: A European perspective. Proceedings of the AREMA 2013 Annual Conference.

Ginisty, A., F. Vallée, E. Soubiran and V.D. Tchapet-Nya, 2014. Report on the railway use-case of the crystal project: Objectives and progress. In: Bondavalli, A. *et al*. (Eds.), SAFECOMP 2014 Workshop. LNCS 8696, Springer International Publishing, Switzerland, pp: 130-136.

Góngora, H.G.C., M. Ferrogalini and C. Moreau, 2015. How to Boost Product Line Engineering with MBSE-a Case Study of a Rolling Stock Product Line. In: Boulanger, F. *et al*. (Eds.), Complex Systems Design and Management. Springer International Publishing, Switzerland, pp: 239-256.

Gopinathan, H., A. Tocci, F. Ciambra and D. Frisoni, 2012. 3.6. 1 Rapid prototyping and validation of human factors model in a model based naval systems engineering application. INCOSE Int. Sympos., 22(1): 448-460.

Gorbea, C., T. Spielmannleitner, U. Lindemann and E. Fricke, 2008. Analysis of hybrid vehicle architectures using multiple domain matrices. Proceeding of the DSM 10th International DSM Conference. Stockholm, Sweden.

Han, S.Y., J.U. Kim and Y.C. Choi, 2012. A study on the development of systems engineering application model for LRT based on MBSE. J. Korea Soc. Syst. Eng., 8(1): 9-19.

Han, S., J. Kim and J.G. Ohn, 2014. The proposal for the application of reverse systems engineering to light rail transit project based on the outputs of ISO/IEC 15288 technical processes. Proceeding of IEEE 17th International Conference on Intelligent Transportation Systems (ITSC, 2014).

Lane, J.A., 2009. Impacts of system of system management strategies on system of system capability engineering effort. Ph.D. Thesis, University of Southern California.

Lee, J.Y., J.K. Ahn, Y.W. Park, W.D. Lee and J.K. Mok, 2002. 6.2. 3 Application of computer-aided systems engineering to develop Automated Train Control (ATC) system. INCOSE Int. Sympos., 12(1): 310-315.

Matar, M., H. Osman, M. Georgy, A. Abou-Zeid and M. El-Said, 2014. Evaluation of Civil Infrastructure Sustainability: A Model-based Systems Engineering (MBSE) Approach. In: Mahdavi, A., B. Martens and R. Scherer (Eds.), ECPPM 2014 eWork and eBusiness in Architecture, Engineering and Construction. CRC Press, Leiden, Netherlands, pp: 327-334.

Pimmler, T.U. and S.D. Eppinger, 1994. Integration analysis of product decompositions. Proceeding of ASME Design Theory and Methodology Conference. Minneapolis, MN.

Prokhorova, Y. and E. Troubitsyna, 2013. A survey of safety-oriented model-driven and formal development approaches. Int. J. Critical Comput. Based Syst., 4(2): 93-118.

Rushton, G. and A. Zakarian, 2000. Modular vehicle architectures: A systems approach. Proceeding of the 10th Annual International Symposium of the International Council on Systems Engineering. Minneapolis, MN, July 16-20, pp: 29-35.

Sage, A.P. and C.L. Lynch, 1998. Systems integration and architecting: An overview of principles, practices and perspectives. Syst. Eng., 1(3): 176-227.

Sage, A.P. and W.B. Rouse, 2009. Handbook of Systems Engineering and Management. John Wiley and Sons, New York.

Scheeren, I. and C.E. Pereira, 2014. Combining model-based systems engineering, simulation and domain engineering in the development of industrial automation systems: Industrial case study. Proceeding of the IEEE 17th International Symposium on Object/Component/Service-Oriented Real-time Distributed Computing (ISORC), pp: 40-47.

Shamsuzzoha, A., P.T. Helo and T. Kekale, 2010. Application of modularity in world automotive industries: A literature analysis. Int. J. Automot. Techn. Manage., 10(4): 361-377.

Smith, R.P. and S.D. Eppinger, 1997. A predictive model of sequential iteration in engineering design. Manage. Sci., 43(8): 1104-1120.

Soubiran, E. and F. Belmonte, 2012. A model based approach for safety analysis. Séminaire sûreté de fonctionnement LIX 26 Octobre 2012, pp: 50-63.

Tadros, M.S., 2013. Integrating the human element into the systems engineering process and MBSE methodology. Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States), Sandia Report, Sand2013-10742, pp: 98.