Establishment of Model-Based Systems Engineering Approach for Rapid Transit System Operation Analysis

Mohamad Azman Bin Othman, Tan Chuan Heng and Oh Sin Hin

Abstract Model-Based Systems Engineering (MBSE) for Rapid Transit System (RTS) applications has recently been adopted by several established railway system suppliers. However, the application of MBSE at the System-of-Systems (SoS) level remains less prevalent in the transportation industry. The adoption of MBSE using Systems Modelling Language (SysML) in this area would offers better management of SoS complexity, enhanced communications among stakeholders and greater support for RTS operators leveraging on models across the systems' lifecycle.

1 Introduction

Designing and building a modern and driverless Mass Rapid Transit (MRT) system has become increasingly challenging today as it involves a huge number of supporting systems across multiple engineering domains (mechanical, electrical, civil and environmental), tightly coupled to one another. Figure 1 shows a typical MRT system, expressed in a top level view together with its main supporting systems.

The successful operation of the MRT system depends heavily on the proper execution of these systems' functionalities at an individual level, as well as at an integrated level. These systems are intricately intertwined so as to achieve the desired emergent properties and performance that consistently meet high standards of reliability, availability and safety.

M.A.B. Othman (✉) · T.C. Heng · O.S. Hin
Land Transport Authority, 1 Hampshire Road, Singapore 219438, Singapore
e-mail: Mohamad_Azman_Othman@lta.gov.sg

T.C. Heng
e-mail: Tan_Chuan_Heng@lta.gov.sg

O.S. Hin
e-mail: Oh_Sin_Hin@lta.gov.sg

M.-A. Cardin et al. (eds.), *Complex Systems Design & Management Asia*,
Advances in Intelligent Systems and Computing 426,
DOI 10.1007/978-3-319-29643-2_12

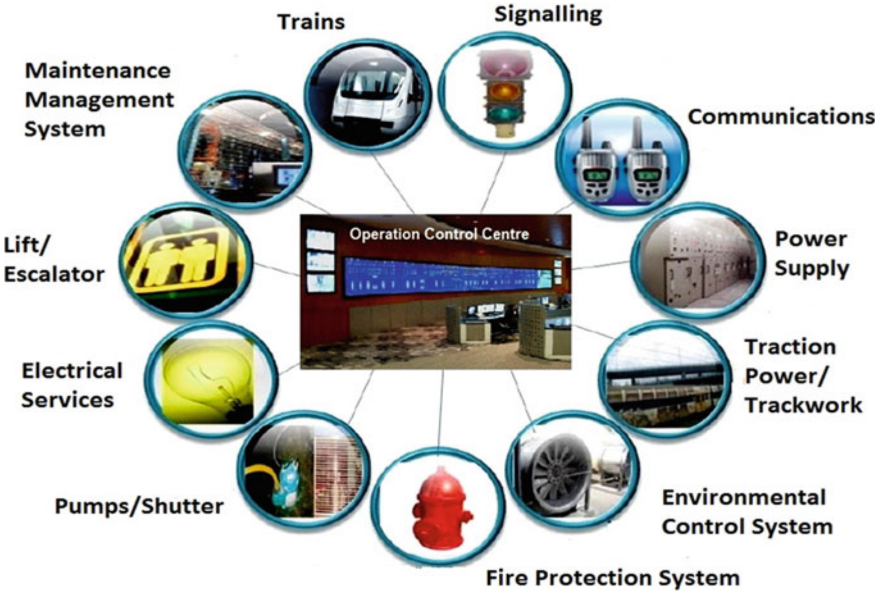


Fig. 1 The composition of a modern MRT

However, the involvement and integration of such multi-disciplinary systems increase the design complexity of a MRT system. Project teams managing the construction of a MRT system not only have to closely monitor the development and testing of individual supporting system, they will also need to understand the design and behaviour of the MRT system as a whole, when these supporting systems are integrated to achieve its desired operation needs.

In addition, due to increasing demand for the expansion of transportation infrastructure, project teams face many challenges to explore new methodology in order to optimise the time required for the design and delivery of a new MRT system. Any ambiguity or uncertainty during the design phase may result in increased costs and prolonged project delivery.

In the past, document-based approach was employed for previous RTS projects. This approach involved manual requirements elicitation, creating drawings manually and independently, producing static diagrams with stored views and processes that depends on manual checking and updating to ensure consistency. This resulted in much difficulty when managing a complex system such as a MRT system.

To overcome such uncertainties, MBSE has been identified as a suitable approach that the project teams can adopt to manage the enormous complexities in their work, especially in the Systems Integration (SI) domain. In addition, with the utilisation of MBSE, the following objectives could be realised:

- To enhance communications among stakeholders (operators, contractors, regulators, asset owner, external agencies and passengers)

- To improve quality of work
- To increase work productivity
- To reduce development risks

In this paper, formulating an approach towards implementing MBSE for automated driverless MRT projects is shared.

1.1 Implementation Consideration

Through the utilisation of SysML, models and diagrams are constructed to provide an accurate and coherent view on the system design and operation of a RTS. System requirements could be elicited, analysed and decomposed to lower level details, facilitating the partitioning and allocation of functions to the respective stakeholders. Also, models that are developed can be linked and traced back to their requirements, ensuring consistency, completeness and correctness of the system design.

Using a RTS project as a reference, a tool based on SysML (IBM Rational Rhapsody®) has been explored to implement models representing various aspects of the RTS, such as operations concept, SoS architecture and operational scenarios depicting various operating conditions.

However, recognising that individual system modeller has different techniques and preferences, there is a possibility that these diversities in design styles adopted by the system modellers might result in the inconsistency of the system models at later stages. The immediate challenge is to align the design in a standard development approach, thus ensuring the system models (and diagrams) remain clear and consistent to other stakeholders. To address these concerns, the issue was looked into and a proposal to adopt a common MBSE framework was established.

The common MBSE framework governs the approach and guides the system modellers on the techniques to model a RTS project. In addition, a supporting document will be compiled so as to dictate the general guidelines and best practices in system modelling, and will serve as a reference for other project teams.

2 Approach in Implementing MBSE

The main objective for establishing a common MBSE framework is to develop a common baseline profile based on standard guidelines and best practices in order to ensure clarity and consistency of models and diagrams, enhancing reusability and concurrent models development. System resiliency checks and analytical studies can be easily defined within the context of the common framework, without any modification to the system models.

With the MBSE framework, two outcomes were sought after. First, the framework layout (embedded with the baseline profile) should be simple and concise, so that any third party would be able to easily access the models through the package descriptions and understand the whole or part of the RTS model based on their interest. Second, a baseline profile will be established in Rational Rhapsody® environment. This profile could then be loaded into individual system modeller's workstation before the start of the project. This will ensure that the start off is from a common baseline and there would be no deviations in the model tool settings.

2.1 MBSE Framework

With the above considerations in mind, the proposed MBSE framework is shown below in Fig. 2.

The highest level consists of Project_TEL_Model (thereafter defined as the SoS) to depict the modelling of SI works on Thomson-East Coast Line (TEL). The system model will be broken down further into eight sub-packages namely: *Structure*, *Behaviour*, *Use Cases*, *Requirements*, *Parametrics*, *Viewpoints*, *IO Definitions* and *Value Types*.

2.1.1 Operation Concept and System Requirements

Firstly, use cases were used to define the SoS and its boundary. Use case diagrams were used to develop models that illustrate the possible operational scenarios that the SoS can be used. These use case diagrams are consolidated under the *Use Cases*

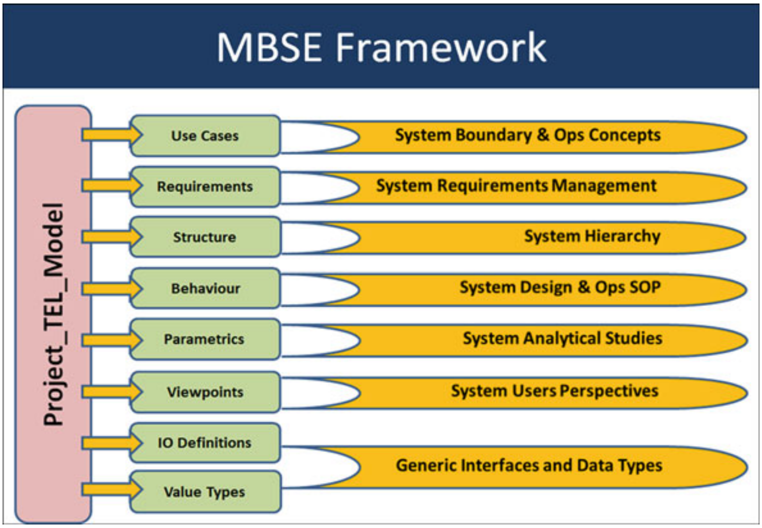


Fig. 2 MBSE framework

package to facilitate better readability from stakeholders who are keen to understand the operational concepts only, without going through the technical details. This will be further elaborated in Sect. 3.

Next, the source requirements associated with the SoS were established by using another tool (IBM Rational DOORS®) for the management of requirements. These source requirements are loaded into Rational Rhapsody®, under the *Requirements* package to fulfill traceability with use cases. From then on, various aspects of the system design were worked concurrently; such as identifying the relevant system functions to satisfy the requirements, formulating the SoS architecture, establishing the functional components and their interfaces.

2.1.2 System Design and Analysis

Block definition diagrams are utilised to illustrate the overview of the SoS, in which the system components are represented as blocks in these diagrams. Detailed design of the system components are illustrated using internal block diagrams, showing the ports and inter-relationships between the components. These diagrams will be grouped under the *Structure* package, which stakeholders can access the contents easily and understand the architecture, hierarchies and their interconnections without much navigation within the SoS.

A series of diagrams (activity, sequence, state machine) was used to illustrate actions, message exchanges, state transitions and events by the stakeholders and/or systems. As these diagrams represent the behaviours of the SoS, they are jointly consolidated under the *Behaviours* package. This package was produced to provide the functional breakdown of the SoS and their detailed message interactions and timings between the system components.

SoS parameters and constraints are illustrated by parametric diagrams in SysML and they are grouped under the *Parametrics* package. The package also consolidates various analytical initiatives, such as performance studies, impact analysis, and design trade-off studies. By accessing the *Parametrics* package, the ability to assess the performance aspects, regulate and monitor system variables without affecting the core design of the SoS is realised.

2.1.3 Stakeholders' View on the System

Individual stakeholder has a different perspective view of the SoS, for example the Station Operator will be more focused on the station aspect as compared to the Traffic Control Operator who will be concerned on train regulation at the Operations Control Centre (OCC). The framework consists of a *Viewpoints* package, with the purpose of framing these perspective views into a common location. This package will contain all views from stakeholders in their respective area of responsibilities (OCC/Train/Station/Depot) and their "linked" activities.

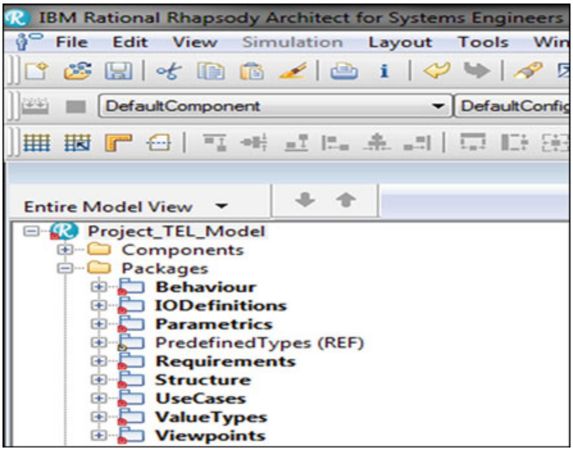


Fig. 3 Baseline profile in rational rhapsody®

2.1.4 Generic System Interfaces

The *IO Definitions* and the *Value Types* packages contain generic system interfaces and data types that are typically used in RTS projects. These are defined based on international standards, in-house engineering standards, best practices and past projects' references. Through these packages, generic models packages are built and these models can be re-used in future RTS projects.

System models developed will be categorised under these sub-packages respectively. In this manner, the simplified layout allows the re-usability, readability and concurrent models development to take place. The baseline profile is developed using the Rational Rhapsody® tool, as shown in Fig. 3.

In parallel to the MBSE Framework, a list of guidelines will be defined to assist the system modellers in their design work. This is to ensure that their works are aligned and maintained at a high level of consistency, thus improving the readability of the system models. As such, a document, *Best Practices of SysML Modelling* was created with the following objectives:

- To provide general system modelling design principles
- To adopt common/well-known terminology in RTS domain
- To adopt a standardised layout convention for various models and diagrams
- To adopt a standardised colour convention for various model elements.

3 Operational Analysis Using MBSE Approach

Adopting MBSE in analysing the rail operation requirement provides benefits that cannot be achieved through conventional document-based approach. The MBSE approach requires the stakeholders involved to be identified and their relationships to be defined in the early stage of development. This results in a more specific and accurate outcome of the study performed, which then translates to a RTS operation model that truly reflects its intended purpose.

Using various SysML diagrams, a clear representation of the relationships and behaviours of the SoS can be portrayed effectively. Within the same project, a single system component declared and used in different diagrams will ensure that traceability can be viewed in the model throughout the system life cycle. Also, this presents an opportunity for impact and trade-off analysis to be performed as any modification to that component will generate a corresponding record of repercussions in the model.

The ability to simulate the operation models offers a glimpse of how the actual system will behave under stressed (e.g. maximum capacity) or degraded conditions (e.g. component failure) and thus exposing inherent system vulnerabilities. In this way, it checks the resiliency of the SoS and proposes improvements such as preventive or corrective mitigations to improve the robustness of the RTS.

During the concept stage, identifying the stakeholders and their needs is the priority in taking a step in the right direction. The high level stakeholders' needs and relationships can be represented using a use case diagram as shown in Fig. 4. The stakeholders are identified as "Actors" while the packages representing the operation regions are placed inside the RTS boundary box. Use cases are placed at appropriate regions to represent stakeholder's needs and association.

Proceeding to design stage, the stakeholders' needs can be further classified into second level behaviour or activities required by the SoS during normal, degraded and emergency mode of operation as shown in Fig. 5. The stakeholders, in particular the operator, are also expanded further to elaborate the actual positions or designations in the organisation that have direct interactions with the SoS.

By analysing a specific scenario for normal, degraded or emergency mode of operation, the actions required by different "Actors" can be described for the normal day-to-day activity or to return the SoS to normal state when faced with an unexpected scenario. Using activity diagrams (Fig. 6), all involved "Actors" are arranged horizontally in a swim lane diagram. The activity box in that lane indicates the action required by the particular "Actor" to perform or to decide at a certain point of the opearation scenario. This will provide clear responsibilities on who should do what, when the need arises or how the SoS should behave in different scenarios.

The activity diagram has provision "call behaviour" for a nested scenario to link up related scenarios when certain conditions warrant for it. Clicking on this box will invoke the function to call up the appropriate scenario. This avoids repetitive creation of the same set of activities from appearing in different diagrams.

Notice that in Fig. 7, the activity boxes are "linked" to various functions of the SoS. These functions are identified by their unique function numbers (e.g. FO2.2.1). These linkages from the activity boxes are required for the down tracing of required functions that are supporting the activity. These activity boxes can also be hyperlinked to the related function for linking purposes and ease of viewing.

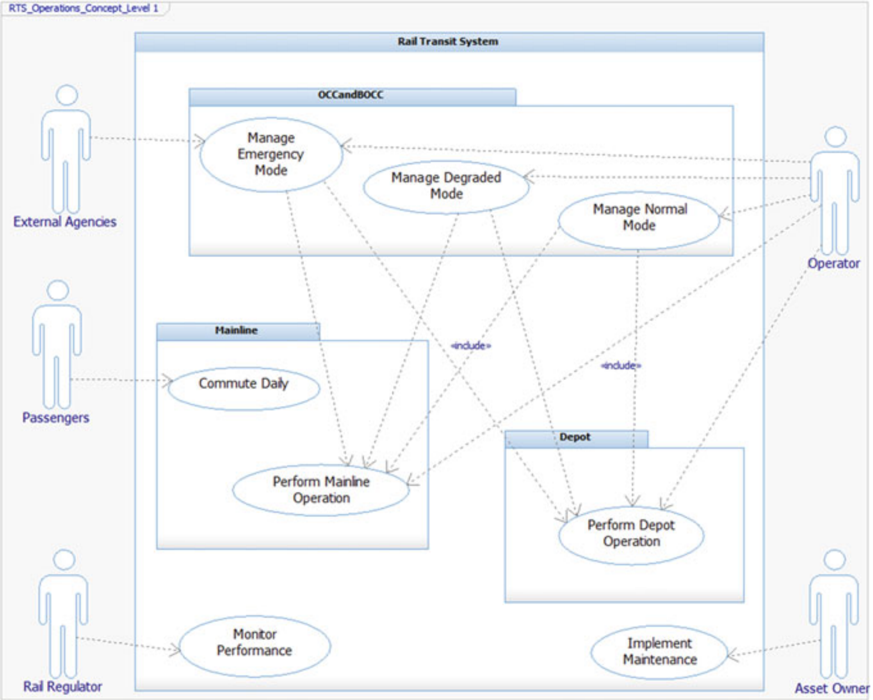


Fig. 4 High level use case diagram

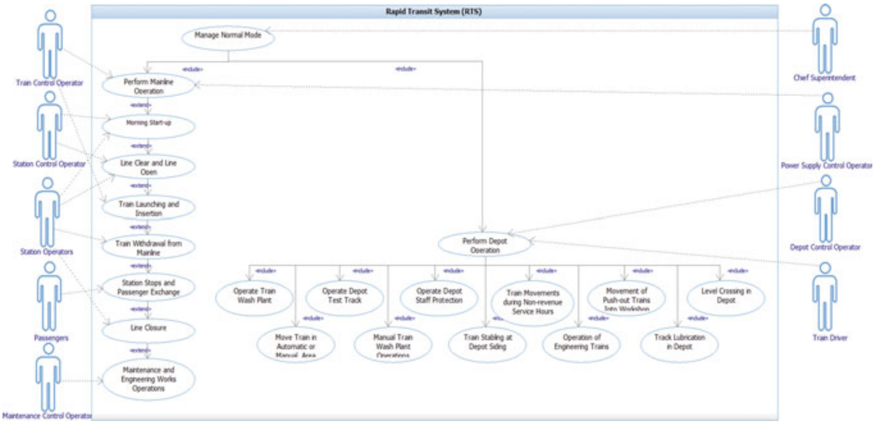

Rapid Transit System (RTS)
Manage Normal Mode
«include»
«include»
Train Control Operator
Perform Mainline Operation
«extend»
Morning Start-up
Station Control Operator
Line Clear and Line Open
Train Launching and Insertion
Station Operators
Train Withdrawal from Mainline
Station Stops and Passenger Exchange
Passengers
Line Closure
Maintenance and Engineering Works Operations
Maintenance Control Operator
Chief Superintendent
Power Supply Control Operator
Depot Control Operator
Train Driver
Perform Depot Operation
Operate Train Wash Plant
Operate Depot Test Track
Operate Depot Staff Protection
Train Movements during Non-revenue Service Hours
Movement of Push-out Trains Into Workshop
Level Crossing in Depot
Move Train in Automatic or Manual Area
Manual Train Wash Plant Operations
Train Stabling at Depot Siding
Operation of Engineering Trains
Track Lubrication in Depot

Fig. 5 Second level behaviour use case diagram

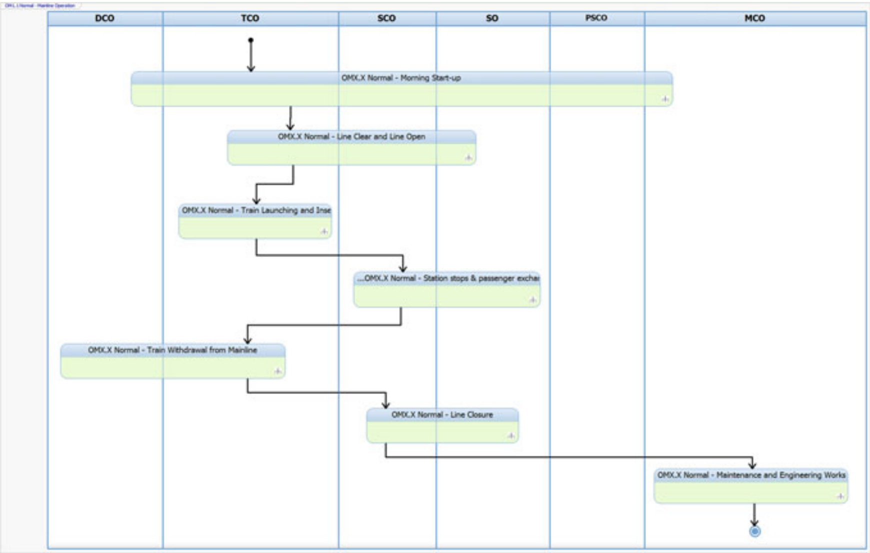
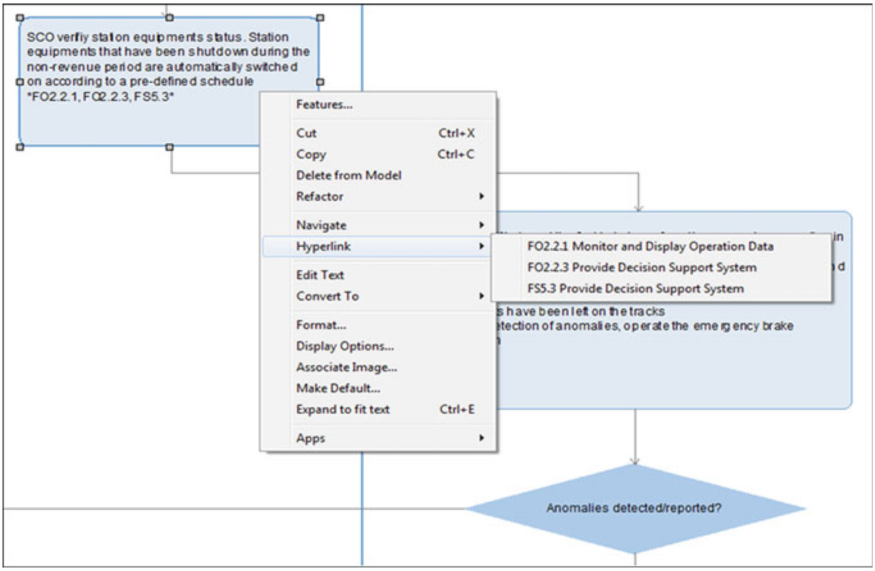

DCO
TCO
SCO
SO
PSCO
MCO
OMX.X Normal - Morning Start-up
OMX.X Normal - Line Clear and Line Open
OMX.X Normal - Train Launching and Inse
...OMX.X Normal - Station stops & passenger exchar
OMX.X Normal - Train Withdrawal from Mainline
OMX.X Normal - Line Closure
OMX.X Normal - Maintenance and Engineering Works

Fig. 6 Representing operational scenario using activity diagram


SCO verfiy stat on equipments status. Station equipments that have been shutdown during the non-revenue period are automatically switched on according to a pre-defined schedule *FO2.2.1, FO2.2.3, FS5.3*

s have been left on the tracks
tection of anomalies, operate the eme rg ency brake

Features...
Cut Ctrl+X
Copy Ctrl+C
Delete from Model
Refactor
Navigate
Hyperlink
Edit Text
Convert To
Format...
Display Options...
Associate Image...
Make Default...
Expand to fit text Ctrl+E
Apps
FO2.2.1 Monitor and Display Operation Data
FO2.2.3 Provide Decision Support System
FS5.3 Provide Decision Support System
Anomalies detected/reported?

Fig. 7 Linking operation activity to functions

4 Model-Based Approach Versus Document-Based Approach

The model-based approach was adopted for a new RTS project, instead of the previous document-based approach. At this juncture, comparison between the two approaches is made to reflect on the improvements achieved with the new approach.

Firstly, using the model-based approach, missing links were able to be identified in design documents submitted by different contractors. Utilising the traceability features provided by the modelling tool, these missing links were uncovered. The cause is due to the usage of multiple link references in various documents and the eventual removal of a base referenced diagram. This problem is prevalent in document-based approach, as reviews of each document could have been done at a different time and there was unawareness on the changes made to the source reference. Maintaining a proper record on the links could be a solution to this but much time would be incurred on updating this record manually.

Secondly, system components that were critical to the SoS were easily identified. Through the system models, information such as systems dependency, quantity and safety-critical connections can be easily derived and represented in a single diagram or chart. Also, there is an additional advantage to develop impact assessment and performance analysis, simply by defining some simulated scenarios based on these system models. This would have taken substantial efforts in the document-based approach, especially when this information exists across several design documents.

Lastly, with the system models created, there is a possibility of reusing these models for future RTS projects. By extracting a base model from existing projects, this can be utilised as a starting point for a new project. With the document-based approach, significant efforts are often incurred in changing the state chart and architecture diagrams manually. This is unproductive and inefficient. Using model-based approach, system models can be reproduced and updated easily to create design documents for the new project. The reduction in time allows system engineers to focus their attention more on system design and analysis.

5 Conclusion

Clearly the MBSE approach offers clarity, traceability, re-usability and coherency of the SoS design and development. Recognising the potential challenges, an approach which consists of the MBSE framework and best practices to assist the system modellers in their area of work was formulated. Through this established approach, the aim is to align the development direction among the system modellers, thus moving towards the ultimate goal of managing system complexities at a SoS level.

Also, a simple comparison was performed between model-based approach and document-based approach. Even though significant efforts were spent on developing the system models, the potential benefits from the model-based approach far outweighs the costs. Exciting possibilities are achievable with the new approach and improvements to productivity level can be expected.

Moving forward, there will be continuous efforts to develop models representing system functions supporting the RTS. The plan is to provide an exhaustive functional analysis model depicting the required high level to lower level functions. Individual function will be allocated to each system for realisation. Subsequently, analysis on the interface or data exchanges between these systems can be carried out by producing a data flow model. Amongst other things, this model should indicate the type of connection (e.g. LAN or hardwired), the type of information (e.g. control or data) and the direction flow of data (i.e. origin to destination). Also, the performance of the interface can be optimised by analysing the information exchanges. This analysis will be carried out progressively in subsequent stages in the future.

References

1. Friedenthal, S., Moore, A., Steiner, R.: A Practical Guide to SysML: The Systems Modeling Language (2012)
2. Hin, O.S., Joyce, H.P.F., Chan, S.: A practical implementation of a generic approach to systems engineering. In: INCOSE Conference (2010)
3. Hin, O.S., Joyce, H., Othman, M.A.: A systems thinking approach to building large-scale land transport systems. Mag. IES Singapore (2015)