Proceedings of CECNet 2022
A.J. Tallón-Ballesteros (Ed.)

doi:10.3233/FAIA220528

Advantage Analysis of MBSE in the Context of Multi-Historical Development Trends

Yuqiang GUO, Shiyan SHE, Jun ZHAO and Junxian GUO[1]
China Helicopter Research and Development Institute, Tianjin 300308, China

Abstract. Since INCOSE first proposed the Model-based System Engineering (MBSE) road map in 2007, the concept of MBSE has been well known, but the application results of MBSE in many companies have not been satisfactory. The implementation of MBSE is a considerable investment, including learning the top-level MBSE processes and the three pillars of MBSE, including the language, tool and methodology. The lack of understanding of MBSE's advantages makes it difficult for many complex equipment development enterprises to invest enormous time and economic costs to implement MBSE. In this paper, the advantages of MBSE are discussed in three developing trends, including written ideograms, human-machine communication language and human thinking. This is the first time to discuss the progressiveness of MBSE from these different views, which can help enterprises dispel the concerns about MBSE implementation and strengthen their confidence in MBSE's full implementation.

Keywords. MBSE, advantage, trend

1. Introduction

INCOSE released the MBSE roadmap (see Figure 1) and its aim was to build a distributed and secure model repository crossing multiple domains in 2020 [1]. But standing at the point of 2022, a large number of enterprises are still stuck in the document-based system engineering approach and have not completed the conversion to model-based system engineering.

[1] Corresponding author: Junxian Guo, General Management Department, China Helicopter Research and Development Institute, Tianjin 300308, China; E-mail: guojx@avic.com.

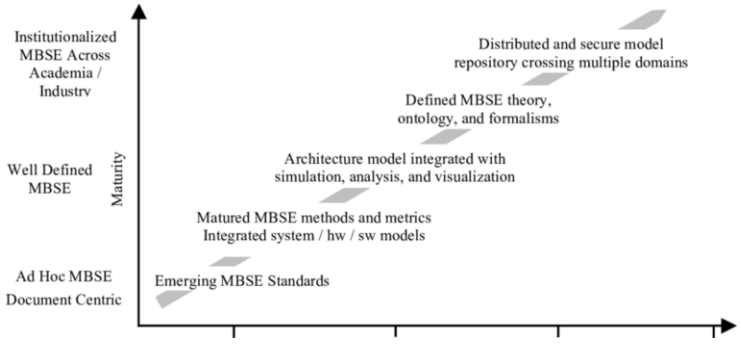

Figure 1. INCOSE MBSE roadmap

Implementing model-based systems engineering requires a breakthrough from learning the top-level system engineering processes described in the INCOSE SE handbook and mastering the three pillars of MBSE (see Figure 2), including the system modeling language, tool, and methodology, which involves a lot of time to learn and practice [2,3].

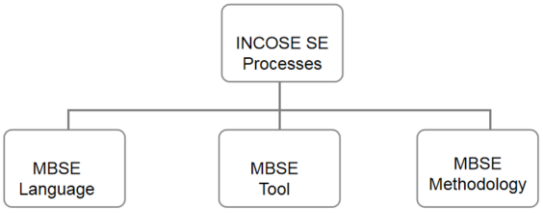

Figure 2. Three pillars of MBSE

Lack of understanding of the advantage of MBSE makes it difficult for companies to invest such a considerable amount of time and economic costs to carry out the implementation of MBSE. For this reason, this article systematically explores the advantages of MBSE from the perspective of the development trends of human written ideograms, computer programming languages and the way humans think. The second part of the article discusses the development trend of human writing and explores the SysML modelling language as an advanced language. The third part discusses the development of human-computer interaction language, i.e., computer programming language. It discusses that SysML is an object-oriented graphical modelling language with the potential for model-driven design and coding. The fourth part discusses the way humans think and figure out that MBSE is the application of modern complex systems thinking, based on recent computer and information technology, which uses multiple model views to build, check and manage system elements and the complexity between them. In the end, the article concludes with a summary.

This is the first time to discuss the progressiveness of MBSE from these different views and it can help enterprises dispel the concerns about MBSE implementation and strengthen their confidence in MBSE's full implementation.

2. The advantage analysis of MBSE from the view of human written ideograms development trend

SysML is the modeling language of MBSE. It is a graphical language to record and show model information [4,5]. We can analyse the progressiveness of SysML from the view of human written ideograms development trend.

As a carrier of human knowledge, the first stage of human writing evolved from narrative pictures. The first stage of writing was pictographs, as shown in Figure 3. Whether Chinese oracle, ancient Egyptian hieroglyphs, Naxi Dongba Pictographs or early Mesopotamian cuneiform writing belong to this category. This writing stage is clear, simple and easy to read and is a common form of the early development of writing.

Figure 3. Pictograms

The advantage of pictograms is that they are easy to read, but the disadvantage is that they are complicated to write and limited to express. It can describe only what can be seen visually, but it can't do anything for the definition of action and abstract concepts. For this reason, people used the ideographic method and simplified abstraction to express a wider range of concepts in the form of abstract symbols (see Figure 4). Nowadays, except for a part of Chinese characters that still retain the characteristics of pictographs, all other scripts in the world belong to the category of abstract hands. The technical specifications is mainly described by text in the traditional document-based system engineering.

Figure 4. Abstract text

In the next stage, to express the relationship between multiple elements, people began to use tables to classify and organize text information, forming a systematic text matrix so that the expression of information is more organized, readable and retrievable, as shown in Figure 5. Microsoft Word tables, Excel, and computer database technology are all based on the term of tables. In the systems engineering processes, requirements matrices, requirements management databases, N2 diagrams, and requirements traceability matrices are tabular.

First Name	Last Name	Department	Supervisor
Fareed	Awad	Marketing	Jane Doe, Vice-President
Adam	Doe	Custodial	Donna Martin, C.E.O.
Jane	Doe	Executive	Donna Martin, C.E.O.
Donna	Martin	Executive	None
John	Smith	Marketing	Jane Doe, Vice-President

Figure 5. Using table technology to organize and store textual information

Given the ambiguity and duality of words, textual descriptions cannot meet the needs of engineering communication exchange, so people began to use logic diagrams to express ideas, such as structure diagrams and flow charts (see Figure 6).

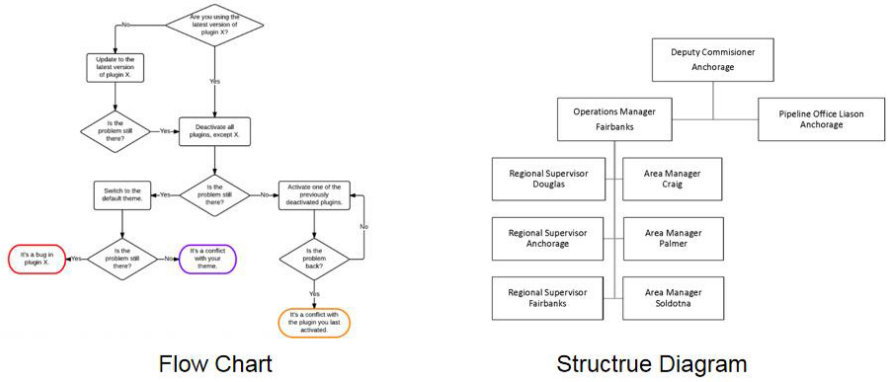

Figure 6. Logic diagram

MBSE uses graphical representations (both tabular and textual) to input and output information (see Figure 7), a formal logical description of the model, which can be read by a computer and transferred on the Internet based on a uniform format. It is an advanced form of ideogram development.

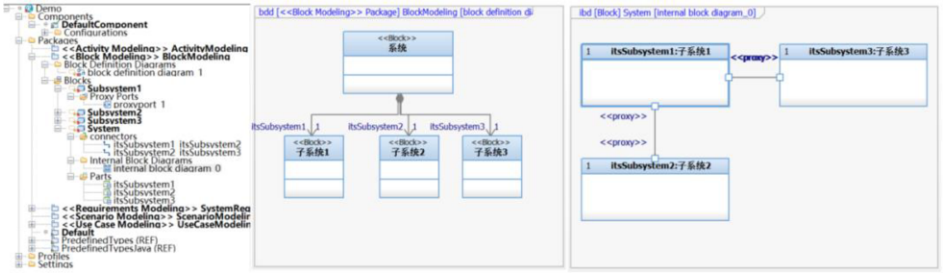

Figure 7. MBSE formal logic diagram

Combined with the above analysis, the development trend of human written ideograms can be shown in Figure 8.

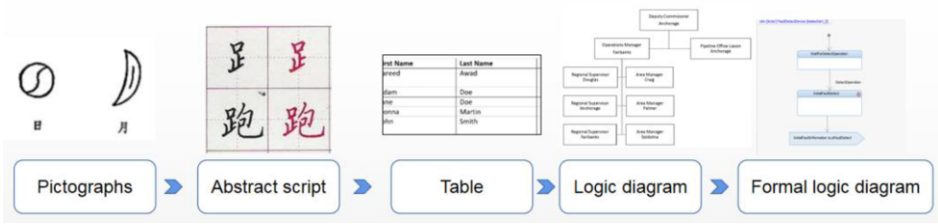

Figure 8. The development trend of written ideograms

3. The advantage analysis of MBSE from the view of human-machine interaction language development trend

In the advanced application of MBSE, SysML model can be transmitted into coding directly [6]. SysML can be read by computer and is also a programming language. In the second part, we can analyse the progressiveness of SysML from the view of human-machine interaction language development trend.

As the language of human-machine interaction, computer programming languages have likewise undergone several developments. When computers were born, people used binary perforated cards to program computer operations, which was the most primitive form of programming, as shown in Figure 9.

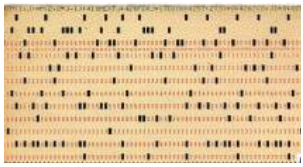

Figure 9. Early perforated card binary programming method

To facilitate communication between people and machines, the assembly language was born as a text-based programming language (see Figure 10), and computer programming began to enter the era of programming languages [7]. Assembly language can be compiled into binary code by a compiler and executed with high efficiency.

```
data segment
string db 10 dup(?),0dh, 0ah,'$'
data ends

code segment
assume cs:code, ds:data
begin:mov ax,data
        mov ds,ax
        mov ah,0ah
        int 21h
        lea dx,string
        mov ah,09h
        int 21h
        mov ah,4ch
        int 21h
        code ends
end begin
```

Figure 10. Text-based programming language

Although assembly language is more efficient in code execution, they are poorly readable and not easy to study and master. A process-oriented programming language, such as C, was invented to solve this problem, as shown in Figure 11. Since then, programming languages have become widely accepted, and computer programs have begun to thrive. Most operating systems are now based on C, balancing readability and execution efficiency.

```
#include <windows.h>
using namespace std;

int main(int argc, char **argv) {

	// shellcode generated by msfvenom
	char shellcode[] = "\xfc\xe8\x82\x00\x00\x00...";

	// allocate space in the process using VirtualAlloc
	void *exec = VirtualAlloc(0, sizeof shellcode, MEM_COMMIT,
PAGE_EXECUTE_READWRITE);

	//copy the shellcode into the allocated space
	memcpy(exec, shellcode, sizeof shellcode);

	//execute the written memory
	((void(*)())exec)();

	return 0;
}
```

Figure 11. Process-oriented programming language

With the arrival of the information society, people's demand for software has exploded. To improve programming efficiency, object-oriented programming languages have been developed, and the famous Java and C++ belong to this category. Object-oriented programming languages encapsulate individuals into objects, describe their properties, and improve the reusability of code with the ideas of inheritance and derivation, thus significantly improving the efficiency of programming, as shown in Figure 12.

```
public class Employees{

//method without parameter
public  void MethodOverloading(){
System.out.println("anything here");
}
//same name but has a parameter
public  void MethodOverloading(int a){
System.out.println("integer a = "+a);
}
//same name but has 2 parameters
public  void MethodOverloading(double b, int c){
System.out.println("double b is = "+b+"integer c is ="+c);
}
}
```

Figure 12. Object-oriented programming language

To improve the readability of code, model-based software engineering was invented, and UML was born. Using powerful programming tools, software code can be displayed in reverse as UML models, making it easy for teams to communicate about the code. On the other hand, UML can be used to generate code directly. SysML was born based on UML and is used for more general system modelling; they are based on object-oriented concepts and are a graphical programming language (see Figure 13), which can be considered an advanced form of programming language development [8].

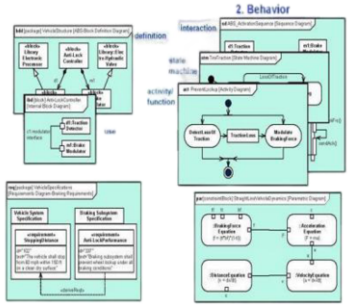

Figure 13. Object-oriented graphical programming language

Combined with the above analysis, the development trend of programming languages can be summarized in Figure 14.

Figure 14. Programming language development patterns

4. The advantage analysis of MBSE from the view of human thinking way development trend

In different historical periods, the evolution of systems thinking can be distinguished into four distinct stages(see Figure 15): the ancient holistic systems thinking, the modern mechanical systems thinking, the dialectical systems thinking, and the modern complex systems thinking, and the fourth is the most advanced way of thinking at present [3].

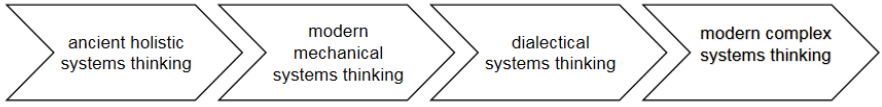

Figure 15. Developmental trend of human systems thinking

MBSE is precisely the application of modern complex systems thinking. It is based on recent computer and information technology, using multiple model views to build, check and manage system elements and the complexity between them [9]. And the SysML modelling language provides a unified modelling paradigm for multiple technical domains such as software, hardware, test, security and reliability, etc. Joint simulation and verification based on the unified paradigm at the early stage of system

development avoids design risks in advance, reduces development cost and time, and enhances the overall system design capability of the R&D teams.

5. Conclusion

To strengthen the determination of enterprises to firmly promote MBSE, the advantage of MBSE is elaborated in-depth. This paper innovatively analyzes the progressiveness of MBSE from the development trends of written ideograms, human-machine communication language and human thinking. It is pointed out that MBSE is an advanced engineering method from the different view analysis. MBSE is progressive and deserves time and fund to perform in the complex system development.

References

[1]. International Council on System Engineering. Systems Engineering Vision 2020, v2. 03. September (2007).
[2]. Ying L I, Sun K W, Yang J, et al. Model-based system configuration approach for Internetware[J]. Science China, 2013(08):1-20.
[3]. Lenny Delligatti. SysML Distilled: A Brief Guide to the Systems Modeling Language. Upper Saddle River, NJ: Addison-Wesley, (2013).
[4]. Maeika, D., & Butleris, R. . Integrating security requirements engineering into mbse: profile and guidelines. Security and Communication Networks, 1-12, (2020).
[5]. Aurelijus Morkevicius. MagicGrid Book of Knowledge: A Practical Guide to Systems Modeling Using MagicGrid from No Magic. Kaunas: Vitae Litera, (2018).
[6]. Pascal Roques. Systems Architecture Modeling with the Arcadia Method: A Practical Guide to Capella. Oxford: Elsevier, (2018).
[7]. Hans-Peter Hoffmann. Systems Engineering Best Practices with the Rational Solution for Systems and Software Engineering. New York: IBM Corporation, (2013).
[8]. International Council on System Engineering. Systems Engineering Handbook: A guide for system life cycle processes and activities,v4.0.San Diego: Wiley, (2015).
[9]. Aditya Akundi and Viviana Lopez. A Review on Application of Model-Based Systems Engineering to Manufacturing and Production Engineering Systems. Procedia Computer Science,185:101–108, (2021).