

IDENTIFYING SUITABILITY OF ANY FORMAL LANGUAGE FOR PARTICULAR APPLICATION AREA

Monika Dhariwal & Shivani Goel

Formal specification of a system has been an active area of research since past few decades. There are a number of formal specification languages, Z, VDM, OCL, SDL [1, 2, 3, 4,] etc. Comparing the different specification methods is not an easy task but it is meaningful in many aspects. This article focus which formal specification language is suitable for particular type of problem e.g. communication type problems, real time application, and problems involving concurrency etc.

Keywords: Formal Specification Language, Reusability, Concurrency.

1. INTRODUCTION

One of the major problems with the software system is the inadequacy of the system and software specification. These specifications are written in a document known as System Requirement Specifications (SRS) [5]. There are software development methods based on graphical techniques, such as data-flow diagrams, finite state machines, and entity relationship diagrams which are proved helpful in developing better specifications, but these lack precisions in details of the specification. The formal specification methods overcome these problems. They specify the system precisely, and provide a smooth way from specification through design to implementation. There are a number of formal specification languages. Any formal language has its own characteristics as well as limitations. In this paper, we discuss characteristics of four formal specification language theoretically, which are the Z language, OCL, SDL, and VDM and their suitability for particular application area.

2. RELATED WORK

The formal languages have been compared on priority of non-functional quality attributes i.e. friendliness, reusability, implementability etc.

Formal specification languages are differ from one another by having different properties like—process-oriented or sequential-oriented or it may be a model-oriented or a property-oriented, and what type of mathematics basis it uses in the following way: Z notation is sequential-oriented, property-oriented, use the set theory and logics. VDM is process-oriented, model-oriented, and use the set theory and logics. Larch is sequential-oriented, property-

Table 1
Comparison of Formal Languages on Priority of non-functional Quality Attributes

Methods Criteria	VDM	Z	SDL
Rigor	3	2	3
Data Modeling	3	3	2
Functional Modeling	3	3	2
Control Structures	2	0	2
Exception Handling	2	0	2
Validity	3	1	2
Modularity	2	2	2
Reusability	2	1	2
Level of Abstraction	3	3	2
Implementability	2	1	3
Friendliness	1	2	3
Tool Maturity	3	1	3

oriented, and uses algebra and logics. Clear is sequential-oriented, property-oriented, and uses algebra. OBJ is sequential-oriented, property-oriented, and uses algebra and logics.

3. PROPOSED WORK

Use of formal language is increasing day by day. There are many formal languages available, but all of them can't satisfy the requirements for formally specifying all type of problems. Any formal language has its own characteristics as well as limitations. All problems do have its own characteristics. So the problem is how to select a formal language that is suitable for the given problem for specifying it formally in a complete and efficient manner with ease. Various problems are studied to categories them to a certain key characteristic. Then the formal language suitable for specification of that type of problem is to be chosen.

* Department of Computer Science and Engineering, Thapar University, Patiala, India.
E-mail: dhariwal.monika@gmail.com, E-mail: shivani@thapar.edu

3.1 Categorization of Problems

1. LSB (Local Switch Board)

The LSB purpose is to connect users locally, with each other [6]. A user interacts with the LSB through his/her phone. The LSB is a simple one, where phone lines and phone numbers are allocated statically at the system startup in a permanent way.

Key Characteristics–Concurrency, Time.

2. Clinical Cyclotron Control System (CCCS)

The Clinical Neutron Therapy System at the University of Washington is a cyclotron and radiation therapy facility that provides cancer treatments with fast neutrons, production of medical isotopes, and physics experiments [7]. The control system handles over one thousand input and output signals. The cyclotron control programs are the control subsystem dedicated to assisting the cyclotron operator.

Key Characteristics–Timing Constraint and Concurrency.

3. Wall Climbing Robot (WCR)

A wall-climbing robot (WCR) is currently under development at Universiti Teknologi Malaysia [8]. The WCR can be categorized as a small-scale embedded hard real-time system.

Key Characteristics–Concurrency, Temporal behavior.

4. ABM (Automated Banking Machine) SYSTEM

Automated banking machine (ABM) having a magnetic stripe reader for reading an ABM card, a customer console (keyboard and display) for interaction with the customer, a slot for depositing envelopes, a dispenser for cash, a printer for printing customer receipts, and a key operated switch to allow an operator to restock money in the machine [9].

Key characteristics–Readability, and inheritance.

5. ChattaBox

ChattaBox, allowed users to communicate via voice, as well as several other features. The ChattaBox requirements were specified using UML use case diagrams, which specifically allow for engineering of system requirements [10].

Key characteristics–Performance, usability, Communication.

3.2 Formal Language(s) Corresponds to Type of Problem :

There are a number of formal languages, but none of them can define completely the specifications in an efficient way.

In this paper we identifying the most suitable language for the particular application area.

Table 2
Suitability of Formal Languages for a Particular Application Area

Type of Problem	Formal Language(s) suitable	Formal Language(s) not suitable & why
Communication	SDL	OCL Reason : It is possible to define a class that can process a set of signal; we have no appropriate way to specify which signals it could send and what consequences are if more than one class can process the same signal. It is even not possible to specify structural containment, in the way as SDL allow it.
Data Modeling, such as Data base system and Purely reactive system i.e. Event driven	SDL	OCL Reason : Since the event driven system the behaviour can be described by using State Machine, a good approach for fully specifying their behaviour. So we prefer to use SDL, Because OCL is not very good at capturing communication.
Application having Inheritance and Polymorphism properties	OCL, SDL	Z Reason: Since inheritance is one of the charters tics of SDL and OCL basically is the extension extension of UML which uses the concept if OOP. But in Z, there is no prescribe way for specifying inheritance.
Time interval properties type application	SDL	Z, VDM Reason: They do not provide any build-in way to express the passage of time.
Concurrency	SDL, Petri Nets.	Z, VDM Reason : They do not provide any build-in way to express concurrency. However, they do not preclude describing concurrent activities. Difficulties arise only when concurrent operations must interact (Clinical Cyclotron Control System)
Predefined Data System	SDL	OCL Reason: Since OCL is just allowing some constraint on UML class diagram and UML has no data system. Also there is a lack of basic data type in data packages and provides no support for collection definition or other type of data definition whereas SLD support a whole data type and data type definition mechanism.
Pure mathematical problem (which uses more mathematical symbols and functions)	Z	SDL, VDM Reason : Since SDL uses only abstract data type and use process and block for specifications. There is no in-built data type for partial functions in SDL and VDM.

4. CONCLUSION & FUTURE SCOPE

After analyzing the characteristics of formal languages and problems, formal languages which are most suitable in different scenarios are specified. This will help in choosing the right formal language for complete specification of a given problem. Characteristics identified are: **SDL** is suitable for problem involving communication, inheritance, concurrency and real-time related problems. **OCL** is suitable for problem involving inheritance, more readability, and reusability. **Z** is suitable for problems involving more mathematical symbols like partial functions, domain & range of a function, small embedded real-time application. **VDM** is suitable for security applications, mathematical properties like mapping etc.

In future, the applicability of more formal languages (i.e. LARCH, HOL, and B-Method etc.) can be studied.

REFERENCES

- [1] [Spi92] J. M. Spivey: “*The Z Notation-A Reference Manual*”, 2nd edition. Prentice Hall, (1992).
- [2] C. B. Jones, “*Systematic Software Development using VDM*”, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, (1990).
- [3] Warmer, J., and A. Kleppe “*Object Constraint Language*”, edition-Wesley (1998).
- [4] Zhang Yaoxue, Chen Hua, Zhang Yue and Liu Guoli, “*SDL-Tran-An Interactive Generator for Formal Description Language SDL*”, Department of Computer Science & Technology, Tsinghua University, Beijing, (1993).
- [5] M. Osborne and C. K. MacNish, “*Processing Natural Language Software Requirement Specifications*”, Proceedings of the Second International Conference on Requirements Engineering, 15-18 April.
- [6] Philippe Leblanc, Ileana Ober, “*Comparative Case Study in SDL and UML*”.
- [7] Jonathan Jacky, “*Formal Specification for a Clinical Cyclotron Control System*”.
- [8] Radziah Mohamad, Dyg. Norhayati ABG. Jawawi, Safaai Deris and Rosbi Mamat, “*Formal Specification of a Wall-Climbing Robot using Z-A Case Study of Small-Scale Embedded Hard Real-Time System*”.
- [9] Munina Yusufu, Gulina Yusufu, “*Comparison of Software Specification Methods using a Case Study*”.
- [10] P. S. Kritzinger, M. Chetty, J. Landman, M. Marconi and Ryndina, “*Chatta Box: A Case Study in Using UML and SDL for Engineering Concurrent Communicating Software Systems*”.