SIG on Formal Methods

NEWS LETTER VOLUME 1 , JULY 2014

SIG on Formal Methods:

Formal Methods (FM) has been in existence since 1940's, when AlanTuring proved the logical analysis of sequential programs using the properties of program states; and Floyd, Hoare and Naur used axiomatic techniques to prove program correctness against the specifications in 1960's.

These initial successes helped inculcate an interest in applying FM to the field of computer science.

Academia has been instrumental in bringing this field to the forefront, through continued research and development. The use of Formal Methods requires an expert skill-set expertise and therefore, its use is limited to those trained in the field.

Mission Statement:

Computer Society of India wants the field of Formal Methods to have a wider audience and more people to benefit from the application of these methods to all spheres of life. There is a need to use effective, correct and reliable approaches to design, develop and qualify complex, high assurance system software with the rigid schedules and budget. For this we need advanced tools, techniques and methods. Industry standards like RTCA DO-178C (Civil Aerospace), ISO 26262 (Automotive), IEC 61513(Nuclear), EN50126 (Railways) have recommended the usage of formal –method based approach to be used in the various phases of engineering process to achieve the required levels of safety and security.

Today there are proven techniques and tools that can be used in specification, design and verification & validation phases to assure correct requirement-capture, implementation, software functionality and security. This helps in developing high assurance software for applications such as cyber-physical systems, net-centric warfare systems, autonomous robots and Next Generation Air Transportation.

- **One day workshop on FM was conducted during April 2013**

Objectives of the Special Interest Group (SIG) are:

- To bring together scientists, academicians active in the field of formal methods and willing to exchange their experience in the industrial usage of formal methods
- To coordinate efforts in the transfer of formal methods technology and knowledge to industry
- To promote research and development for the improvement of formal methods and tools with respect to their usage in industry.
- To bring out practical engineering methods where formal methods will be integrated with current engineering methods

Some of the known applications of formal methods are:

- Formal verification, including theorem proving, model checking, and static analysis
- Techniques and algorithms for scaling formal methods
- Use of formal methods in automated software engineering and testing
- Model-based formal development
- Formal program synthesis
- Formal approaches to fault tolerance
- Use of formal methods in safety cases
- Use of formal methods in human-machine interaction analysis
- Use of formal methods in compiler validation and object code verification

3. Committee Members

1. Ms. Bhanumathi K S, Convener
2. Mr.Chander Mannar
3. Prof.Anirban basu
4. Mr. Suman Kumar
5. Prof. Shyam Sundar
6. Ms. Manju Nanda
7. Ms. J. Jayanthi
8. Ms. Saroja Devi
9. Prof. Meenakshi D'Souza
10. Yoganand Jeepu
11. Dr. Swatnalatha Rao

Planned Activities:

National Work Shop and Conference on Formal Methods for Software Engineering
Venue: I I Sc , Bangalore, October 12-16, 2014

Convener:

Bhanumathi K S
“Ganadhakshya” #406, 8 C Main,
H R B R First Block
Kalyan Nagar
Bangalore 560043
Email:bhanushekar@gmail.com
Mobile:+91 95350 92589

Choosing a Formal Method

Courtesy and Source: Formal Methods Europe FME Homepage

SCOPE

This document is aimed at those who have decided that it would be a good idea for their project or organization to use a formal method, but is not sure what steps to take next: which formal method or which kind of formal method will be best for their purposes.

In order to result in a successful outcome, choosing a formal method requires the same approach as any other technical decision. First, one needs to be clear about one's objectives and to know what are the technical, organizational and management constraints. Then one may focus on the characteristics of a formal method which will meet those objectives and constraints. Finally a method can be selected which most closely conforms to, and whose tools, experience and support most closely conforms to those characteristics.

GENERAL APPROACH

This guide is a framework which, we hope, will help you to construct for yourself a decision process tailored to your own application and organization. The sections which follow consist of possible aims, criteria, characteristics and needs. Considering whether each of these applies should help to focus on and define the objectives, constraints and other criteria particular to an application and organization which form the context of a choice of formal method, i.e. your application and that part of your organization which will be using a proposed formal method. The bulleted points, which can be treated as a set of questions for you to ask yourselves, are a guide only; they can be extended or modified according to each individual organizational and technical environment. What is important is to establish and define that environment as clearly as possible before proceeding to make a selection. The purpose of the questions is to help the user organization to determine appropriate criteria. Accordingly, the questions are grouped into some principal categories in the following sections:

- General reasons for choosing formality
- Characteristics of the application
- Criteria for success of application
- Needs and constraints of the organization

The answers to these questions form a context in which you can then focus on what are the characteristics of the formal method which is most suitable for your needs. To help you to do that, the "possible characteristics" of a formal method are presented in the last main section:

- Characteristics of a formal method

Having gone through this process you can then match your list of requirements against what is on offer, talking to tool suppliers etc.

GENERAL REASONS FOR CHOOSING FORMALITY

Software and system quality, consistency and integrity can be improved by formalizing different products and processes in the development cycle. Are the reasons you want to apply formal techniques:

- To improve quality and rigor of whole development process?

This would be the case if your organization wished to adopt a formal approach to software or system development as part of a general improvement of its development process. An improvement of development technology, like adopting formal methods, is a valid aspect of process improvement just as are improvements in documentation, configuration management, measurement etc.

- To improve integrity, reliability or other characteristics of the system?

In some circumstances formal methods may be applied to system development as well as software development. The analysis and design of the system architecture usually precedes the specification of software components. Do you propose to apply formal methods to the development of the larger system as well as the software?

- To reduce specification errors?

Expressing the specification, particularly the functional specification, of software and system components helps greatly to reduce errors in specifications. Is this your principal motivation?

- To improve requirements definition?

Requirements, especially system requirements, are usually expressed in non-formal language. The process of deriving formal expressions of specifications from them nearly always has the effect of improving those requirements definitions. Experience indicates that omissions and inconsistencies in the requirements statements are found more reliably than with other techniques. Is requirements analysis and definition the phase of the life-cycle which you particularly wish to address?

- To improve documentation and understanding of designs?

There is at present a great quantity of legacy software which is undocumented or with very inadequate documentation. Maintaining the software is as a result extremely difficult and confidence in its reliability is low. Good documentation reduces these problems and formal descriptions of the life-cycle products dramatically reduce them. Is this your motivation for using formal methods?

- To provide a firmer foundation for maintenance and enhancement? See above.
- To explore the properties of a design architecture?

In some applications it is not possible to characterize the behavior of the context of the software; in some telecommunications environments, for example, patterns of traffic may be to an extent unpredictable. A formal model of the design of the software can provide an understanding of its properties and limitations. Do you wish to improve your knowledge of your software's properties by more accurately modeling the design?

- To provide a more rational basis for choosing test data?

Relatively recently, techniques have been developed for deriving test data from functional specifications of software components. Such test sets may be more closely related to the specifications which the software is designed to fulfill, and the process of test set derivation is more systematic if formal methods are used for functional specification.

- To be as certain as possible that the design and implementation are error-free?

In safety-critical and other critical contexts it can be of utmost importance to ensure that the software is functionally free from errors. Applying formal descriptions to specifications and designs enables proofs of correctness between those successive stages of the development cycle.

- To meet particular customer or standards requirements?

Some contracts mandate the use of formal methods during certain stages of the development cycle, or mandate adherence to standards which do so.

*****