# SIG on Formal Methods

NEWS LETTER VOLUME 2 , AUGUST 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 13-15, 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

# *Applying Formal Methods in Software Development*

Compiled by Bhanumathi K S

In computer science, specifically software engineering and hardware engineering, **FORMAL METHODS** are a particular kind of mathematically based techniques for the specification, development and verification of software and hardware systems. Our society is increasingly dependent on computer systems. As a result, mistakes in these computer systems can have dire consequences: large monetary damage can occur, or human lives can be endangered. In these cases, the traditional error prevention methods are often insufficient. However, employing formal methods can help to achieve the desired safety level.

Formal methods in software engineering are mathematical techniques that are used in the design, implementation and testing of computer systems. The application of mathematical methods in the development and verification of software is very labor intensive, and thus expensive. Therefore, it is not feasible to check all the wanted properties of a complete computer program in detail. It is more cost effective to first determine what the crucial components of the software are. These parts can then be isolated and studied in detail by creating mathematical models of these sections and verifying them.

In order to reason about mathematical models, several different techniques have been developed- **MODEL CHECKING AND THEOREM PROVING**.

The first method takes a finite transition system and systematically checks whether all the desired properties hold for every state of the system. Because the number of states increases exponentially with the size of the model, this method will often have to limit itself to small variants of the system that is under investigation. An advantage of model checking is that, when it finds a problem, it can indicate how it got into an error state. This information can be used to improve the model.

The second method uses general mathematical techniques to reason about the models. This makes it possible to reason about systems of which the number states is unlimited. For this a price must be paid: the reasoning cannot occur fully automatically. By combining the strong points of model checking; automation and finding counter examples, with the more general mathematical power of theorem proving, it takes less effort to guarantee the reliability of the investigated systems.

****