SIG on Formal Methods

NEWS LETTER VOLUME 1, JANUARY 2015

SIG on Formal Methods:

Formal Methods (FM) has been in existence since 1940's, when Alan Turing 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. Dr. Yoganand Jeepu
11. Dr. Swatnalatha Rao
12. Dr. Aditya Kanade
13. Dr. A. Indira
14. Mr. Dhinakaran Pillai

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

Formal Specification

Compiled by Bhanumathi K S

What is a specification?

A specification is a description of a product (either to be build or existing). Specifications are used in many different engineering disciplines including software engineering. In software engineering the products that are specified are software. Associated with the notion of a

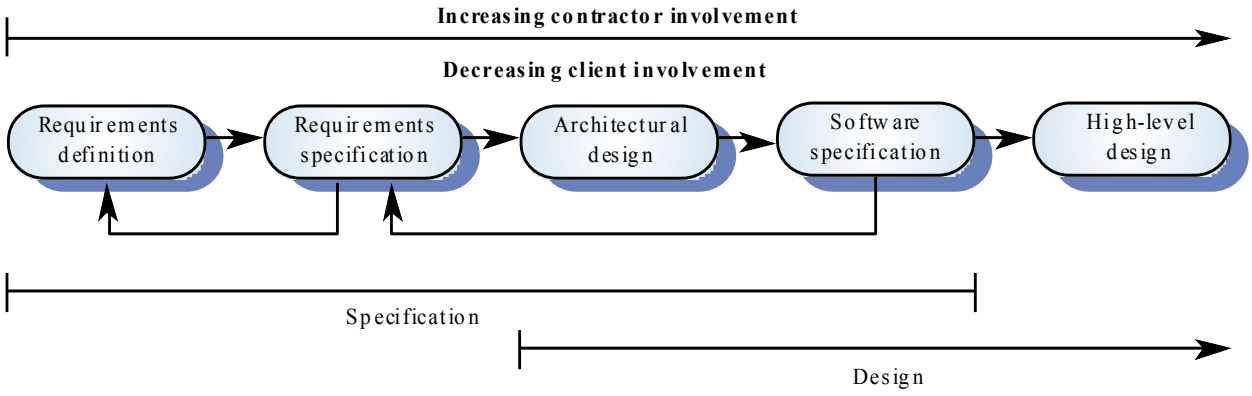
specification, there is the notion of what it means for a product to satisfy (fulfill/meet/conform to/be compliant with) its specification.

In computer Science, formal specifications are mathematically based techniques whose purpose is to help with the implementation of systems and software. They are used to describe a system, to analyze its behavior, and to aid in its design by verifying key properties of interest through rigorous and effective reasoning tools.

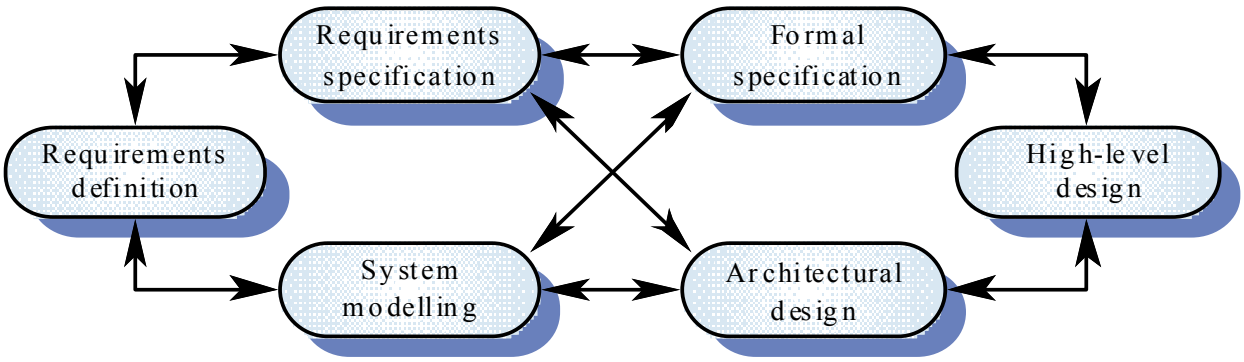
These specifications are *formal* in the sense that they have syntax, their semantics fall within one domain, and they are able to be used to infer useful information. These are all based on mathematical representation and analysis of software

Specification and design are inextricably mixed. Architectural design is essential to structure a specification. Formal specifications are expressed in a mathematical notation with precisely defined vocabulary, syntax and semantics

Specification and Design:



Specification in the software process:



Specification Technique:

Algebraic approach:

- The system is specified in terms of its operations and their relationships

Model-based approach:

- The system is specified in terms of a state model that is constructed using mathematical constructs such as sets and sequences.
- Operations are defined by modifications to the system's state.

Use of formal Specification:

Formal specification involves investing more effort in the early phases of software development. This reduces requirements errors as it forces a detailed analysis of the requirements incompleteness and inconsistencies can be discovered and resolved.Hence, savings as made as the amount of rework due to requirements problems is reduced

Interface Specification:

Large systems are decomposed into subsystems with well-defined interfaces between these subsystems. Specification of subsystem interfaces allows independent development of the different subsystems. Interfaces may be defined as abstract data types or object classes.

The algebraic approach to formal specification is particularly well-suited to interface specification.

Behavioral specification:

Algebraic specification can be cumbersome when the object operations are not independent of the object state.

Model-based specification exposes the system state and defines the operations in terms of changes to that state.

Formal Specification can be used as contracts or communication media between customer and developers, and between developers. Besides being used as a base for design and implementation, formal specifications can also be used as a base for generating test cases, for simulation and for formal analysis of the described products in order to predict their behaviour before they are implemented.

****