Formal methods

A Factfile provided by the Institution of Engineering and Technology
About This Factfile
The Institution of Engineering and Technology acts as a voice for the engineering and technology professions by providing independent, reliable and factual information to the public and policy makers. This Factfile aims to provide an accessible guide to current technologies and scientific facts of interest to the public.

For more Position Statements and Factfiles on engineering and technology topics please visit http://www.theiet.org/factfiles.

The Institution of Engineering and Technology
The Institution of Engineering and Technology (IET) is a global organisation, with over 150,000 members representing a vast range of engineering and technology fields. Our primary aims are to provide a global knowledge network promoting the exchange of ideas and enhance the positive role of science, engineering and technology between business, academia, governments and professional bodies; and to address challenges that face society in the future.

As engineering and technology become increasingly interdisciplinary, global and inclusive, the Institution of Engineering and Technology reflects that progression and welcomes involvement from, and communication between, all sectors of science, engineering and technology.

The Institution of Engineering and Technology is a not for profit organisation, registered as a charity in the UK.

For more information please visit http://www.theiet.org

© The Institution of Engineering and Technology 2011

The Institution of Engineering and Technology is registered as a Charity in England & Wales (no 211014) and Scotland (no SC038698).

Enquiries
policy@theiet.org

Contents
What are Formal Methods? .................................................... 3
What advantages do Formal Methods have? .......................... 3
Why is this important? ........................................................... 3
What Formal Methods are widely used? ................................. 3
Who uses Formal Methods for real systems? .......................... 3
Where can I find more information? ........................................ 3
End Notes ............................................................................ 3
Useful Websites ..................................................................... 3
What are Formal Methods?
“Formal Methods” are well-founded techniques, frequently supported by tools, for systems and software development. Their distinguishing characteristic is that they exploit notations that have a mathematically rigorous semantics so that the analysis tools available can offer high levels of assurance and consistency in the analysis of models, designs and programs.

What advantages do Formal Methods have over more traditional methods?
Formal Methods enable the software or hardware developer to specify their system rigorously and to analyse it to ensure that the specification is consistent and reasonably complete and that any system that meets the specification is certain to have the required properties.

Formal Methods then help to de-risk the development process by reducing the likelihood of late detection of defects or by reducing the cost of defect detection. The static analysis techniques embodied in FM tools, including model checking and formal proof, can be deployed on models produced in early development stages to identify areas of incompleteness and ambiguity in requirements and specifications. When applied to late design or code-level analysis, they can be used to identify specific patterns of defect such as bad memory or pointer management.

Formal Methods even enable the developer to prove mathematically that the system they have developed has particular, critical properties (for example, that it conforms to its security requirements).

Why is this important?
It is not possible to show that a software system is free of errors by testing it, because testing every possible path through the software, for every combination of data that could possibly cause that path to fail, would take an impossibly long time. Whereas many physical systems (such as components of a bridge or an aircraft wing) obey natural laws that allow the engineer to test a design at extreme values and to infer the behaviour between those extremes, the behaviour of computer systems is a succession of separate states whose relationships depend on the logic of the program and not on physical laws (other than those governing the correct functioning of the microelectronics). It is therefore invalid, in general, to assume that a program functions correctly for some input values it will also function correctly for other input values - even if the latter values lie between the extremes of the former values.

The practical consequence is that software systems almost always contain many errors, even after rigorous testing during development and in user acceptance. Academic and industrial studies have shown that there will typically be between 5 and 30 errors in each thousand lines of program text at the time the software is put into service, after the normal degree of system testing and user acceptance testing.

Where developers have used formal methods, even for a part of their development, it becomes possible for software tools to analyse their work and to detect errors (or show their absence) far more quickly, cheaply and thoroughly than conventional methods can, in the same way that a compiler for a strongly-typed programming language can detect and eliminate many classes of programming error.

Formal Methods can also support the automation of testing (test generation, test execution, and checking the test output). This is a form of ‘model based testing’ but with the added rigour that can only be provided through a formal semantics. This approach has been used in telecoms, some safety critical systems, and in Microsoft for testing parts of Windows. Formal model-based testing can show significant savings when compared with manual testing.

What Formal Methods are widely used?
There are many Formal Methods used commercially and industrially. VDM, Z, B, Event-B, SCADE, SPARK, Promela and the SPIN model-checker are a few examples mentioned in the survey of industrial application of formal methods cited below. This survey included systems ranging from one thousand to one million lines of code.

Who uses Formal Methods for real systems?
In hardware design, the major chip designers such as Intel. In software development, major companies such as Boeing, Siemens, Airbus, BAE Systems, Rolls Royce, SAP, Microsoft and IBM. There are many consultancies in the UK, France, Germany, Scandinavia, the USA and elsewhere that provide commercial training, support, development staff and software tools.

Where can I find more information?
- A range of excellent introductory material can be found on the Formal Methods wiki at http://formalmethods.wikia.com/wiki/Publications#Introduction.

End Notes
This content can contribute towards your Continuing Professional Development (CPD) as part of the IET’s CPD Monitoring Scheme. http://www.theiet.org/cpd