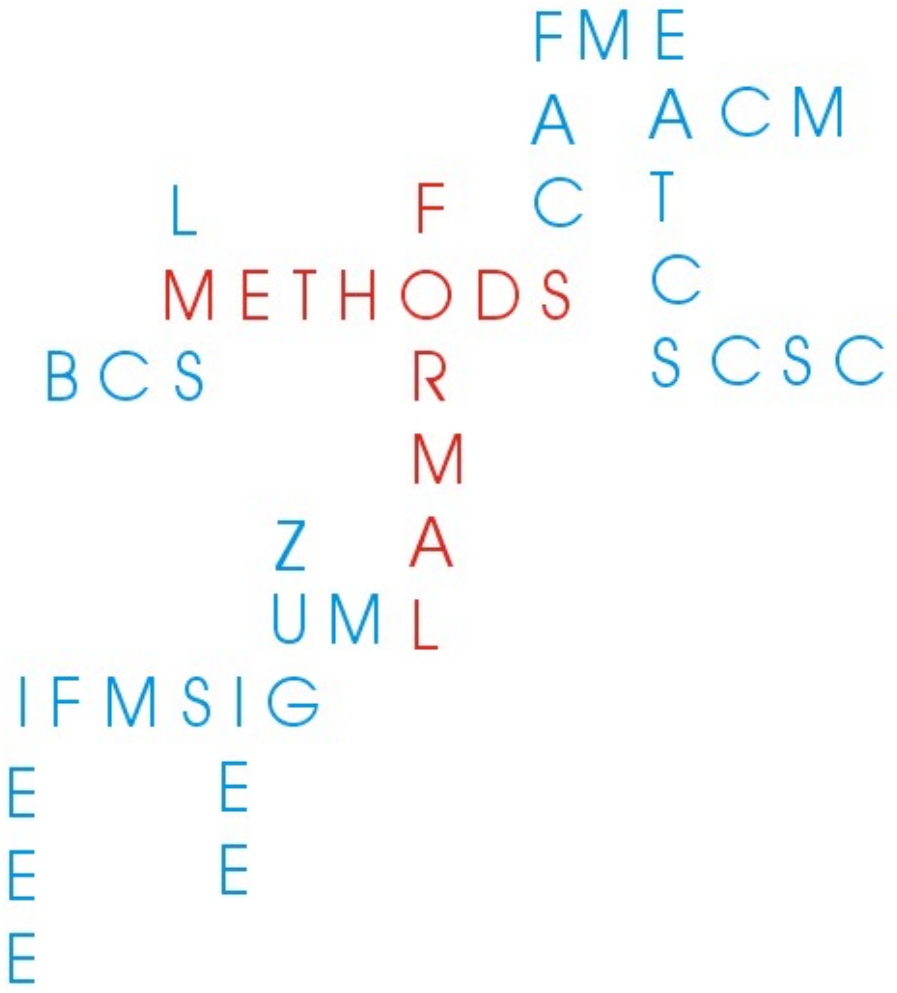
Issue 2009-1
July 2009

FACS A C T S



BCS FACS

The Newsletter of the Formal Aspects of Computing Science (FACS) Specialist Group

About *FACS FACTS*

***FACS FACTS* [ISSN: 0950-1231] is the newsletter of the BCS Specialist Group on Formal Aspects of Computing Science (FACS). *FACS FACTS* is distributed in electronic form to all FACS members.**

Submissions to FACS FACTS are always welcome. Please visit the newsletter area of the FACS website [http://www.bcs-facs.org/newsletter] for further details.

Back issues of *FACS FACTS* are available to download from:

http://www.bcs-facs.org/newsletter/facsfactsarchive.html

The *FACS FACTS* Team

Newsletter Editor **Margaret West [m.m.west@hud.ac.uk]**

Editorial Team **Paul Boca, Jonathan Bowen, Jawed Siddiqi**

Contributors to this Issue

Paul Boca, Jonathan Bowen, Tim Denvir, John Fitzgerald, Anthony Hall, Jawed Siddiqi, Margaret West

If you have any questions about FACS, please send these to Paul Boca [paul.boca@googlemail.com]

Peter John Landin (1930–2009)

It is with great sadness that we note the death of Peter Landin on June 3rd 2009. Peter was a major contributor to Computer Science in general, and to semantics and functional programming in particular. An obituary will be published in the next issue. Readers with personal recollections of Peter are invited to contact the editor so that these can also be included.

Editorial

Some brief news follows of our activities this year and proposed activities for next year. Our FACS evening seminar series commenced as usual in autumn 2008. The first of these, a joint seminar with the London Mathematical Society (LMS), was given by John Tucker (University of Wales, Swansea) on November 11th 2008. The next seminar was given by Jos Baeten (Eindhoven University of Technology) on April 2nd and was held jointly with *Formal Methods Europe*. Short reports on these are presented later in the Newsletter. The last seminar was given by Mark Harmon on 26th May and concluded with a panel session. The Christmas Meeting took place at the BCS London Office on 9th December 2008: Formal Aspects of Safety Critical Systems and was a joint event with the Safety Critical Systems Club and, again, with Formal Methods Europe. A report, written by Jonathan Bowen, is presented later in this Newsletter. Previous seminars have been written up by their speakers and will appear towards the end of the year in the form of a book published by Springer. We will let you know when it has been published.

Some committee news – Rick Thomas has resigned his role as liaison with the London Mathematical Society. Thanks are due to Rick who has ably performed his role and helped us in the organisation of some stimulating joint seminars. These have taken place in the interesting environs of the LMS HQ, De Morgan House, a Grade II listed building in Russell Square, Bloomsbury. Thanks are also due to Tom Melham for accepting the liaison role and for helping to arrange our next joint event.

For 2009–2010, we have so far planned two seminars, the first in collaboration with BCS Women and taking place on October 19th 2009 at the BCS London Offices. It will be given by Professor Marta Kwiatkowska (http://web.comlab.ox.ac.uk/people/Marta.Kwiatkowska/) of the Oxford University Computing Laboratory, the title and abstract to be given later. The second seminar is to be delivered by Professor Mike Gordon FRS (http://www.cl.cam.ac.uk/~mjcg/) of the University of Cambridge Computer

Laboratory and entitled *Forward with Hoare.* The seminar is jointly held with the LMS and will take place on December 1st 2009 at De Morgan House in London.

You will no doubt be aware that the BCS is undergoing a period of transformation. You will have already had some indication of the implications in a recent copy of *IT Now.* More details will be provided in the autumn. One of these changes involves the scheduling of specialist groups Annual General Meetings which have previously taken place in spring and early summer. These have now been moved to the autumn and we will let you know the date of our AGM when it has been finalised.

Formal Aspects of Safety-Critical Systems
FACS Christmas Meeting – 9 December 2008

Report by Jonathan P. Bowen

Martyn Thomas

Chris Dale

Paul Boca

On 9th December 2008, BCS-FACS joined forces with the Safety Critical Systems Club to co-organize a one-day meeting on the *Formal Aspects of Safety-Critical Systems* at the Strand Palace Hotel in London. The event was also supported by the Centre for Software Reliability at Newcastle University and Formal Methods Europe.

The first speaker was Martyn Thomas, a very well established figure in this field, who spoke on the *Assurance of sociotechnical systems*. He emphasised the role and fallibility of humans, musing on the part that formal methods can play in this light. He argued that the cost of using formal methods compared to traditional approaches can often still be greater for the degree of dependability that is required at present, but as this increases in the future, formal methods will become comparatively cheaper for the dependability that will be expected.

Other speakers gave more detailed information on the use of formal methods applied to safety-critical systems from both academic and industrial viewpoints. A highlight was the experience of the use of formal notation and tools at Praxis High Integrity Systems, a leading company in this area, presented by Rod Chapman. The company especially uses the Z notation for formal specification and SPARK Ada for implementation, often in combination. He emphasised the need for better tools and especially the need for proper support for those tools.

John Fitzgerald

Rod Chapman

The Christmas meeting brought together the communities interested in safety-critical systems and formal methods. A special thank you must go to Paul Boca of BCS-FACS, Chris Dale of SCSC, and John Fitzgerald of FME for making this meeting possible. It is hoped that such synergies can continue for the future.

Beware of Horseless Carriages

Tim Denvir

The first cars had an engine at the front, with space for the driver close behind. The driver, like a coachman in charge of a horse, sat behind the engine to operate its controls and held the steering wheel as if it were reins. Behind was the passenger compartment, often closed off from the driver, who could be exposed to the elements. These “horseless carriages” were a simulation of the horse-drawn coach. It took some decades of evolution for car design to focus on the passengers, who are its rationale and motive. Car designers clung to the horse-drawn metaphor and only reluctantly opened their eyes to the new possibilities of internal combustion technology. We seem to be technologically conservative, adapting to new engineering opportunities with lamentable inertia. Do computing and software advances suffer from the same laggardly torpor? I think so. Here are a few more examples.

Before digital computers were widespread, engineers used to pre-test control systems using analogue computers. Units which could perform addition, multiplication, integration and sign inversion were wired together with leads on a patch panel, and the outputs measured in response to input stimuli. Soon the patch panels were made detachable so that the “programs” could be removed and returned again for another run. The engineers were in effect using modular electronics to program sets of simultaneous integral or differential equations. When digital computers arrived, electronic engineers asked for analogue computer simulators which could run on a digital machine. These were easily provided and the programs looked rather like a crude autocode. It would have been quite easy to extend this language to solve integral and differential equations directly, but for many years the engineers preferred to go their accustomed long way round, pretending that they had an analogue computer and then simulating it with a digital machine. I discovered that engineers were still using this approach to design control systems for aero engines in 1989, 25 years after analogue computers were on the way out.

At much the same time I worked for a well-known software house, which was commendably keen on its quality standards. These prescribed the writing, reviewing and signing off of documents at many stages of a product’s development. Although computers were used for preparing and editing the documents, great store was set on signatures as evidence of responsibility and approval. Thus, we were heavily reliant on paper. I proposed that we should move to an entirely computer-based document system with appropriate computer-based authentication, but a manager told me that automatic signature recognition was not practical and was beyond the state of the art. Resisting the temptation to bang my head against the wall, I explained that a computer-generated, secure, unforgeable system of codes could fulfil the same function as a hand-written signature.

A last example: industry stayed with cathode ray tubes for TV sets and computer monitors for a long time after LCDs and other technical methods were

possible. Fortunately, CRTs are now almost completely obsolete, but for a clumsy, heavy device they were remarkable persistent: the earliest form was demonstrated in 1897, before even the thermionic vacuum tube[1].

The moral is: instead of duplicating the actions of a previous technology with a new one, deconstruct it and reconstruct, re-solving the requirements it met, using the latest techniques**.**

So, are we committing the same error, that of designing horseless carriages, with formal methods? Formal semantics, denotational, operational, algebraic, are geared to defining the meaning of linear programming languages, whose sentences consist of a linear scripts of symbols. But what is a programming language for? It enables us to communicate to the computer our desire to build a program. Having to do this by means of a linear, preprepared script must surely be on its way out. Dialogue is the next form of programming language. We already see the beginning of this in spreadsheet and database packages. A spreadsheet will perform linked calculations in a number of cells according to formulae which have been placed there by the "programmer". The development of this takes place through a dialogue and the parts of the "program" form a matrix rather than a linear script. They can be composed in any order. In the future we may well see total dialogue systems where the computer will construct a program in response to a conversation, possibly spoken, questions and answers on both sides, with the "programmer".

So should we not be now devising semantic techniques for defining the meaning of these new kinds of language before they arrive? That way we may influence their design to be semantically coherent, tractable and well-formed. If we don't, we shall be forever retrofitting formality to syntactic systems that have been amateurishly cobbled together.

Now, while I am about it: wheels! Wheels are inefficient, cause energy waste through friction, require bearings which wear out and have to be replaced, and need expensive smooth surfaces to run on. Yet we persist in clinging to this 5,000 year-old technology. There must surely be a better way of getting about...

timdenvir@bcs.org

1 Keller, Peter A.: "The 100th Anniversary of the Cathode-Ray Tube", Information Display, Vol. 13, No. 10, 1997, pp. 28–32.

Reports: Evening Seminar Series 2008–09

The first seminar (which was held jointly with the LMS) was given by John Tucker (University of Swansea) and took place on 11th November 2008 at De Morgan House. The talk commenced with some history, the year 2008 being the 450th anniversary of the death of Robert Recorde (born 1510 in Tenby). Among many other mathematical achievements, Recorde was the originator of the equals symbol `=' which appears in his book The Whetstone of Witte (1557). The commercial revolution in 13th Century Europe resulted in many books of arithmetic being published, and this lead to the concept of `equations for everything'. Over 400 years later Physics World (2004) published the result of a survey of the most highly regarded equations of Physics: Maxwell's Equation was voted top, with Euler's equation second.
(See http://physicsworld.com/cws/article/print/20407.)

The applications of equations to Computer Science, includes the use of equational reasoning in Process Algebra. In addition equational specifications have been used to model the rational numbers; this involves the novel concept of a *meadow* where every integer (including zero) has a restricted form of inverse. The unification of computing and physical systems was then discussed by the speaker, together with the intuitions this could provide for understanding the latter. Questions from the audience included the following: *Can you model performance?* The speaker suggested that *there can be many co-existing models – however these might model only a small part of the system.*

The second evening seminar (which was held jointly with FME) was given by John Baeten (Eindhoven) and took place on 2nd April 2009. The speaker noted that Automata theory and formal language theory is a basic model of computation addressing discrete behaviour of interacting processes/agents. However it was the speaker's contention that there was a lack of a proper treatment of interaction. An abstract model of computation was then presented consisting of interacting links between the external world and finite control, and between finite control and memory. The second abstract model considered was that of the communicating process and the talk discussed integration of the two theories, and the benefits ensuing on both sides.

The talk concluded with a presentation to Dr Nico Plat of "West Consulting" in the Netherlands, who had just retired after ten years of magnificent service as the Secretary of FME and a long period before that when he had served the FME group before it became a legally independent association. Nico is a keen cook and so was presented with a copy of the Larousse Gastronomique (in Dutch!).

Book Announcement

The following book has been suggested as of interest to FACS FACTS readers:

Justifying the Dependability of Computer-based Systems with Applications in Nuclear Engineering by Pierre-Jacques Courtois. The book is in the series: Springer Series in Reliability Engineering with ISBN: 978-1-84800-371-2 and Online version available.

ABSTRACT

What evidence is sufficient to justify the release of a computer-based safety critical system? How should this evidence be presented to certification bodies or regulatory authorities? What best practices should be applied? These are just a few of the questions addressed by *Justifying the Dependability of Computer-based Systems*, which provides a framework for the justification of the dependability of a computer-based system. The book also explores some of the more fundamental aspects of safety evaluation, such as the nature of models, arguments, evidence and documentation, and the ways to deal with different types of risk and uncertainty.

http://www.springer.com/engineering/production+eng/book/978-1-84800-371-2

For more information contact:
Pr. Pierre-Jacques Courtois (pierre-jacques.courtois@uclouvain.be)

Announcing the Z Word Tools

Anthony Hall

Anthony Hall (http://www.anthonyhall.org) has sent the following announcement of the Z Word Tools:

The Z Word tools are a collection of tools for editing, checking and using Z specifications within Microsoft Word. The tools include:

1. styles for laying out schemas and other Z paragraphs;
2. a Unicode font that includes all the Z symbols and is visually compatible with Times New Roman;
3. automatic layout of Z paragraphs like the LaTeX equivalent, with italic text, formatted keywords and so on;
4. the ability to enter symbols from a palette or (for died in the wool LaTeX hackers) typing in the markup;
5. one-click typechecking, with errors highlighted in the Word document;
6. generation of indexes and cross-references to definition and use of Z names;
7. COMING SOON: creation of diagrams showing the specification structure, using Graphviz;
8. the ability to hide the Z completely so the document can be used by maths-phobic readers;
9. miscellaneous tools such as checking matching brackets.

The intention of the tools is:

- to lower the barrier to the uptake of Z by removing at least one obstacle, the need to learn another document production method;
- to allow easy integration of Z with natural language, diagrams, tables and other notations relevant to the domain;
- to encourage incremental development of Z specifications by allowing frequent typechecking;
- to encourage the writing of good natural language by producing documents with the mathematics hidden.

The tools currently use Mike Spivey's fuzz as the underlying typechecking engine. They work by exporting LaTeX mark up and importing the fuzz error report: the intention is that this mechanism could be used with other tools, in particular tools supporting the Z standard. Typechecking does not require declaration before use and works across multiple documents: for example an operations document can be checked against a separate data model document. The tools are freely available from http://sourceforge.net/projects/zwordtools. There is a web site describing the tools at http://zwordtools.sourceforge.net.

FACS FACTS Issues in 2009

Call for Submissions

We welcome contributions for the next issue of *FACS FACTS*, in particular:

- Letters to the Editor
- Conference reports
- Reports on funded projects and initiatives
- Calls for papers
- Workshop announcements
- Seminar announcements
- Formal methods websites of interest
- Abstracts of PhD theses in the formal methods area
- Formal methods anecdotes
- Formal methods activities around the world
- Formal methods success stories
- News from formal methods-related organizations
- Experiences of using formal methods tools
- Novel applications of formal methods
- Technical articles
- Tutorials
- Book announcements
- Book reviews
- Adverts for upcoming conferences
- Job adverts
- Puzzles and light-hearted items

Please send your submissions (in Microsoft Word, LaTeX or plain text) to Margaret West [editor@facsfacts.info], the Newsletter Editor.

If you would like to be an official *FACS FACTS* reporter or a guest columnist, please contact the Editor.

Forthcoming Events

BCS FACS Seminars

Unless stated otherwise, these take place at:
BCS London Offices
First Floor, The Davidson Building
5 Southampton Street
London WC2E 7HA

BCS FACS Seminar — October 19th 2009
Marta Kwiatkowska, Oxford (joint with BCS Women)

BCS FACS Seminar — December 1st 2009
Mike Gordon, Cambridge (joint with London Mathematical Society)
Note different venue: De Morgan House, 57–58 Russell Square, London

FM2009 — November 2nd – 6th 2009
Eindhoven, NL

ABZ 2010 — February 2010
Quebec, Canada

For further conference announcements, please visit the Formal Methods Europe (FME) website [http://www.fmeurope.org], the EATCS website [http://www.eatcs.org] and the Virtual Library Formal Methods website [http://formalmethods.wikia.com/wiki/Meetings]. For general formal methods information, see the Formal Methods Wiki [http://formalmethods.wikia.com].

FACS Committee

Jawed Siddiqi
FACS Chair

Jonathan Bowen
FACS Treasurer
and ZUG Liaison

Paul Boca
Secretary

Roger Carsley
Minutes
Secretary

John Cooke
FAC Journal
Liaison
and BCS Liaison

John Fitzgerald
FME Liaison
SCSC Liaison

Judith Carlton
Industrial Liaison

Margaret West
Newsletter Editor

Tom Melham
LMS Liaison

Mark D'Inverno
Chair, State-based
Specification Subgroup

John Derrick
Chair, Refinement
Subgroup

Rob Hierons
Chair, Formal Methods
and Testing Subgroup

FACS is always interested to hear from its members and keen to recruit additional helpers. Presently we have vacancies for officers to help with fund raising, to liaise with other specialist groups such as the Requirements Engineering group and the European Association for Theoretical Computer Science (EATCS), and to maintain the FACS website. If you are able to help, please contact the FACS Chair, Professor Jawed Siddiqi, at the contact points below:

BCS FACS
c/o Professor Jawed Siddiqi (Chair)
Sheffield Hallam University
E **info@bcs-facs.org.uk**
W **www.bcs-facs.org**

You can also contact the other Committee members via this email address.

Please feel free to discuss any ideas you have for FACS or voice any opinions openly on the FACS mailing list [FACS@jiscmail.ac.uk]. You can also use this list to pose questions and to make contact with other members working in your area. Note: only FACS members can post to the list; archives are accessible to everyone at http://www.jiscmail.ac.uk/lists/facs.html.

Coming Soon in FACS FACTS…

Conference reports

Details of upcoming FACS Evening Seminars

And More…