



The Royal Academy
of Engineering



Engineering Values in IT

A joint study by The Royal Academy of Engineering,
the Institution of Engineering and Technology
and the British Computer Society





The Royal Academy
of Engineering



Engineering Values in IT

A joint study by The Royal Academy of Engineering,
the Institution of Engineering and Technology
and the British Computer Society

© The Royal Academy of Engineering
ISBN 1-903496-97-0
July 2009

Published by
The Royal Academy of Engineering
3 Carlton House Terrace
London
SW1Y 5DG

Tel 020 7766 0600
Fax 020 7930 1549

Copies of this report are available at: www.raeng.org.uk/engineeringIT

Table of Contents

1.	Introduction	3
1.1	The critical role of IT	3
1.2	Study approach	4
2.	Executive summary	5
2.1	Conclusions	5
2.2	Findings and recommendations grouped by stakeholder	6
3.	Engineering and IT	8
3.1	What is engineering? What is a professional engineer?	8
3.2	What makes an IT project? Where does IT benefit from an engineering approach?	8
3.3	IT and design tools	11
3.4	IT and mature engineering processes	12
4.	Engineering skills in IT projects – critical application areas and critical roles	13
4.1	Application areas requiring engineering skills	13
4.2	High consequence and high cost systems: licensing practitioners and recording failures	16
5.	Education and training for success in IT	18
5.1	Education and training	18
5.2	Research in industry and academia	19
6.	Professionalism and ethics in IT	21
6.1	Why is chartered status important for IT professionals?	21
6.2	Qualifying as a Chartered Engineer	21
6.3	Qualifying as a Chartered IT Professional	22
6.4	The need for professional development	23
6.5	Ethics in IT	23
7.	Understanding the barriers – how engineering approaches can be implemented in IT	25
7.1	Confusion between developing and using IT	25
7.2	What is conceivable is not always possible	25
7.3	“It wasn’t tested enough”	26
7.4	Errors should no longer be regarded as inevitable	26
7.5	Science based methods are not impractical	26
8.	Concluding remarks	27
9.	Appendices	28
	Appendix A: Methods	28
	Appendix B: Engineering competences	29
	Appendix C: Software design tools	31
	Appendix D: Case study	33
	Appendix E: Statement of ethical principles	34
	Appendix F: Full survey results	36

1. Introduction

1.1 The critical role of IT

IT systems form a crucial part of the UK's critical national infrastructure. They support transport and communications networks, the health service, banking and retail systems. Every financial transaction using debit or credit cards is processed by an IT system that handles payments worth in the region of €52 billion each day. Road transport is governed by IT systems that monitor traffic and send out instructions to drivers in order to reduce congestion and potentially improve the efficiency and reduce the carbon footprint of motoring. Road vehicles themselves are partially controlled by embedded IT which governs several aspects of vehicle performance, from engine management, to air bag control, to GPS; and the emergence of autonomous vehicles will mean an even greater role for IT in vehicle control. IT systems are increasingly embedded in buildings, controlling sensor-based systems that monitor building use and regulate the heating, cooling and lighting systems efficiently. IT is central to patient care with a great deal of health care technology and the new electronic medical records system dependent on IT systems.

IT also plays a central role in commercial enterprises large and small, from the entertainment industry to computer aided manufacturing. Up-to-date and efficient IT systems are essential to keeping these industries competitive and allowing them to meet the demands of consumers. The IT industry itself makes a highly significant contribution to the UK economy and with professional leadership it can provide a firm foundation for a robust knowledge economy. IT is also central to innovation. The internet is itself both an IT innovation and a platform for further business innovations. IT supports many areas of science, technology and engineering – from the computing power needed to sequence the human genome to the systems that make air travel and space exploration achievable and safe.

This critical role that IT plays across sectors makes it crucial that IT systems are reliable, robust and usable. Past IT failures have in some cases caused significant waste or loss of money and damage to reputation. Such IT failures cannot be tolerated when they are in systems that safeguard the health and safety of the public. Where IT supports vital infrastructure, any vulnerabilities in the software involved will mean vulnerabilities in that infrastructure. Such vulnerabilities are targets for terrorists or hostile states, and significant research and practice needs to be invested in building IT systems that are as robust as possible. This includes investment on behalf of those that procure the IT systems – be they within industry or government.

However, for every IT failure there are many successful IT systems. There has long been an area of IT research and practice devoted to the development of high consequence systems – such as the avionic systems that control aircraft – and the lessons learned in this sector should be adopted in the development of other high-consequence systems particularly those relating to safety, security and high finance. There is significant potential for the area of reliable and dependable computing to develop and grow, and there is a need for it to do so. It is also vital that the professionalism of all those involved in the commissioning or delivery of IT systems develops to match the responsibilities placed upon them. This report addresses the specific issue of the need for professionalism in IT systems development. In particular, it will set out the contribution that engineering skills, capabilities and values can make in addressing IT challenges.

1.2 Study approach

This study was undertaken by a group of Fellows of The Royal Academy of Engineering, the British Computer Society (BCS) and the Institution of Engineering and Technology (IET). The group focussed on how greater professionalism could be brought to bear on the tasks of specifying, procuring or developing software-based IT systems. The group deliberately concentrated exclusively on professionalism in IT systems development, to the exclusion of consideration of the much wider range of managerial, operational and technical issues which impact on IT systems; for consideration of these wider concerns see The Royal Academy of Engineering and BCS report, *The Challenges of Complex IT Systems*.

The study group obtained input from senior professionals in the IT and civil engineering disciplines drawn from industry, government and academia. In addition significant data was gathered by conducting a survey of membership groups within the BCS and IET.

Throughout this report we use the term 'engineer' as a shorthand to describe a professional who designs and develops systems (in any medium or materials) using methods that justify a high degree of confidence that the costs and risks will be controlled and that the resulting system will have the properties required by its users. This description fits all the traditional engineering specialisms (civil, aeronautical, structural, electronic and so on) and our study set out to see whether, and to what extent, the description is also relevant to certain areas of IT, in particular, software development. IT is a wide and varied profession, requiring skills ranging from high level management and strategy to the technical skills of those designing and implementing software. Our concern is with those IT professionals involved in system design, development and implementation, in particular of high-consequence or safety critical systems, and the extent to which engineering methods and skills are important to their work.

When referring to 'government' in this report, we intend to include all government departments and local government, all of which are likely to procure or develop major IT systems. This report should be of particular interest to those in the Civil Service with responsibility for procuring IT systems and for managing the technical staff in their departments.

2. Executive Summary

2.1 Conclusions

The concerns of this report relate to the area of IT dealing with the development of software-based IT systems. This area of IT has benefitted greatly from the input of engineering methods and skills. It is the argument of this report that IT systems engineering has now reached the point of maturity where it is a codified discipline that must be recognised as a required capability for system developers. Like all branches of engineering, IT systems engineering has a basis in scientific theory. It exploits scientifically sound design automation tools, and it uses the concepts of science to model not only the structure and detail of each projected product design, but also the environment of its use and the requirements of its users. Education and training providers, from universities to management schools, must ensure that they keep up to date with current research and understanding in this area in order to provide IT students and professionals with an understanding of the current scientific basis of their discipline. There are strong parallels here with the move of the medical profession in the mid 19th century from a craft to a discipline with practitioners required to be educated in the then newly emerging corpus of medical science.

It is also the argument of this report that IT systems development requires the professional qualities of integrity and ethics essential to all areas of engineering and it requires good professional judgement, exercised in the true interest of the client, to make all necessary compromises between costs, timescales, features, risk and technical feasibility. This report recommends that IT professionals therefore work to achieve chartered status, the standard mark of professionalism and competence. Chartered status, either Chartered Engineer or Chartered IT Professional, requires a commitment to an institution's code of conduct, and to continuing professional development, which involves the broadening and deepening of experience and maintaining an understanding of advances in the underlying science, changes in the legal and social environment and the development of new engineering tools and technology. The rapid pace of progress and development in IT means that IT systems engineers need to be continually developing their professional knowledge and experience.

In order to stimulate uptake of chartered status, procurers of large IT systems in government and industry should employ chartered professionals to lead and manage these projects. This report recommends in particular that appropriate chartered status should be a requirement on leading engineers engaged in development of systems with implications for safety or national security.

Strong leadership within the IT profession will help to ensure that the IT industry continues to grow and to have the success it has enjoyed so far, and that IT is recognised as providing the foundation for most of the UK's critical infrastructure and for the products and services that each of us uses every day.

2.2 Findings and recommendations grouped by stakeholder

Government and industry:

There should be appropriate IT and engineering expertise in Government in order to ensure that it is an intelligent customer; able to ask for and specify the right solutions, to consult with relevant experts and act appropriately on advice received, and to ensure that large-scale IT systems are procured from appropriate sources.

We recommend that Government work together with professional institutions to procure advice on major IT projects, and that the Civil Service aim to recruit competent qualified IT professionals and engineers with appropriate knowledge and experience of IT application areas. (Sections 4.1 and 4.2)

While chartered status is a strong indicator of professionalism in respect of capability and ethical approach, take-up will continue to be limited until it is actively required by employers and those placing IT contracts.

Appropriately qualified Chartered Engineers and Chartered IT Professionals should be employed to lead and manage major IT projects within both Government and industry. This is of particular importance for high-consequence systems where engineering skills must be employed to ensure that systems meet requirements and that risks are controlled. (Sections 4.1, 4.2 and 6.1)

The availability of robust and reliable design and development tools is growing and facilitating the production of certifiably dependable software components. It is important that customers are able to establish the maturity and capabilities of software tools and to have confidence in the dependability of software components.

Industry bodies should work with professional institutions to develop and promote an accreditation scheme that would provide customers with auditable evidence of the dependability, maturity and capability of software tools and components. (Section 3.3)



Universities and training bodies:

Engineering methods are particularly relevant to the most demanding IT projects. Universities should equip students with the specific skills needed to deal with IT projects that pose the highest risk of failure.

Universities should develop their undergraduate education to reflect better the engineering realities of large scale IT developments; e.g. the complexities of embedded systems, the dependability issues for enterprise systems and the fact that most developments involve modifying or interfacing with pre-existing software. (Section 5.1)

Universities, management schools and training providers need to strengthen their continuing professional development programmes to reflect the increasing capability and relevance of the research results being developed in academia and industry in order to ensure that competent professionals are able to deploy the most effective tools and techniques on their problems.

Providers of IT education and training must work with researchers in academia and industry to ensure that students acquire a current, science-based understanding of the development of IT systems. (Section 5.2)

Professional bodies:

An increasing number of IT Systems are 'high consequence systems' in that their failure could have catastrophic consequences. Obvious examples of such systems are to be found in the fields of safety critical systems, high security systems and critical financial systems. At present there are no formal qualifications required of practitioners in these fields. It is recognised that such qualifications would be domain specific (analogous to licence to practice in the inspection of large dams in the civil engineering profession).

It is recommended that the relevant professional institutions, under the guidance of the UK Engineering Council and in conjunction with the Health and Safety Executive, explore an advanced qualification, as an addition to chartered status, for practitioners in high consequence domain areas including safety critical and secure systems (Section 4.2)

IT professionals:

The criticality of IT systems means that those who design and maintain them must be professional in their work. This entails both a commitment to adhering to professional ethical standards and to maintaining expertise. This is especially important to professionals working on high consequence systems where knowledge of the best and most reliable solutions is essential, and in the many areas of IT where technologies are rapidly changing and disruptive technological change is likely.

IT professionals should aim to achieve chartered status and thereby commit to the professional and ethical code of an institution. This entails maintaining professional knowledge and competency through Continuing Professional Development (CPD). (Section 6.4 and 6.5)

In the many areas of IT where the rate of technological change is rapid, professionals have a duty to develop a rigorous understanding of the science underpinning IT systems development alongside a commitment to maintaining awareness of the most up to date tools and technologies. (Sections 6.4 and 6.5)

3. Engineering and IT

In this section we explore the characteristics of engineering that are common to all of the traditional engineering disciplines, and discuss the extent to which IT systems development shares these characteristics.

3.1 What is engineering? What is a professional engineer?

The following is a short definition of the engineering method:

An engineer is someone who can be trusted to design, build, maintain and retire a product that is cost-effective and fit for purpose. Engineers have learnt that it is necessary to:

- (1) make effective use of scientific results, so that their designs can be shown to have the required properties
- (2) use mature engineering processes and appropriate tools, so that the development project is cost-effective and risks are controlled
- (3) use strong quality control and assurance methods, to catch and correct errors early and to provide objective evidence that the product will be fit for purpose
- (4) behave at all times with integrity, so that the customers are protected from unrealistic expectations and unprofessional work.

Engineering brings a creative and innovative approach to problems. Engineers are trained to analyse a client's needs, to offer the best solutions to those needs and to develop a product according to the constraints set by the client. These skills are crucial for engineering projects and depend on both training in the analytic methods of engineering needed to assess design solutions, and the creative abilities nurtured through engineering practice. Experience is essential for an engineer, as seeing how models and plans might diverge from the reality of implementation is a key skill that engineers develop. Also central to engineering practice is risk management. Engineers are skilled in assessing the risks associated with a system and controlling them so that they are kept at a tolerable level.

A more detailed account of the skills needed by professional engineers is given by the UK Engineering Council's (EC^{UK}) UK-SPEC (UK Standard for Professional Engineering Competence). A condensed version of the competences for a professional engineer is given in appendix B.

3.2 What makes an IT project? Where does IT benefit from an engineering approach?

In this report we are comparing IT projects with traditional engineering. By IT projects, we mean the following:

The construction of computer-based systems, starting with the negotiation of requirements taking into account working practices, followed by the analysis of requirements, high-level design, the construction of the system from existing and/or custom-built components, integration with other systems, implementation in the application environment and support and upgrading throughout the service lifetime through to and including the retirement of a system.

What are the comparisons between this sort of project and those in 'traditional' areas of engineering? In a study by The Royal Academy of Engineering and the British Computer Society, *Challenges of Complex IT Systems*¹, it was argued that

Process redesign, change management, testing - people don't perceive that you need an engineer to do these sorts of things. These roles are the key parts of any project or programme, but are not seen to be the preserve of engineers or even skills that engineers have at all. We have the legacy view that engineering is about building bridges and machines. People don't understand the USP of engineering in these fields.
Brian Derry, NHS

¹ *Challenges of Complex IT Systems*, published by The Royal Academy of Engineering, April 2004

complex IT projects need a systems architect. Analogies can be drawn between the role of the architect in an IT project and that of the architect in a building project. Both involve the development of socio-technical systems and both require an understanding of how the technical system being designed is going to serve the people who will use it. This means that both have a significant challenge in elucidating the requirements of a client, and ensuring that the final specification for a project actually meets the need of the client. In both cases, the architect has the role of acting as a translator between the business and technical aspects of a project.

Traditional areas of engineering such as civil engineering and IT projects alike are liable to run into difficulties due to a lack of understanding of the complex needs of users. Thus a social housing project can fail because the vision did not match the reality of how residents really use the space around them and IT projects can fail because they are not sufficiently sensitive to the needs of the users of the system.

IT projects pose some particular challenges that place them with the more demanding areas of engineering. For all socio-technical projects, the elicitation of the requirements is extremely demanding, and many significant IT projects are socio-technical in nature. Unlike structural or civil engineering projects where visual plans can be drawn up and shared, for IT projects there is no well-established formalism for externalising software design. This means that the stakeholders have difficulty in communicating and reasoning about requirements and in determining how achievable a particular design might be (see section 7.2, 'what is conceivable is not always possible').

Whilst many engineering projects are carried out in well-understood situations and have fixed goals and end-dates, IT projects often take place in a more fluid and ever-changing environment (one which is often affected by the IT project itself). Software systems have to be built to evolve, especially as they are often a crucial part of complex business changes. Users' requirements are often extremely complicated because they are tightly matched to business processes. Therefore in IT, as in other areas of engineering, projects do not end with delivery. The whole lifecycle has to be taken into account, from development and delivery, to maintenance, to decommissioning and moving to a new system.

There are also other respects in which IT should resemble areas of engineering. Lessons are learnt from failures in civil and structural engineering in particular, where analysis of failure is a key part of the discipline. Henry Petroski, author of *To Engineer is Human: The Role of Failures in Successful Design*² wrote the following on the failure of the Tacoma Narrows Bridge in Washington:

Subsequent analysis of the Tacoma Narrows failure confirmed that the bridge span acted much like an airplane wing subjected it to uncontrolled turbulence. This aerodynamic aspect of bridge design was one that is no longer welcomed, and susceptible bridges contemporary with the Tacoma Narrows were quickly stiffened against crosswinds with steel that may have ruined their gracefulness but insured their structural safety. Subsequent designs were tested in wind tunnels much the way new airplane designs are. (pp. 164-5)

It is a characteristic of traditional engineering professions that, when a major failure occurs, the root causes are investigated and the lessons are learnt by the whole profession, so that where possible there are no future failures from the

² Henry Petroski, *To Engineer is Human: The Role of Failures in Successful Design*, Vintage, 1992

Engineering skills do pay in that engineers retain learning. In IT, it is more practice based, everything appears to be done for the first time. But there are some areas where the engineering 'cookbook' knowledge can be used – taking IT systems engineered and designed in one place and adapting them for others.

John Suffolk, Government Chief Information Officer

same causes. IT failures should lead to just the same sort of investigation to identify faults and their root causes. Lessons learned from failures should then be implemented in any subsequent similar projects.

Ideally, accumulated knowledge should be incorporated directly in the design tools. There are already examples of this in IT. There is a great deal of research into designing programming languages which are simpler to use and to have inbuilt solutions to common problems. For example in Java standard libraries are built in to avoid deadlocks in concurrent systems. The Erlang language was developed specifically to solve well known problems in telecommunications networks. Such improvements need to become the norm so that, for example, no future product suffers from the weaknesses in programming language design that make buffer overflow security vulnerabilities so commonplace.

Is IT an engineering discipline? Survey findings

79% of 85 respondents considered themselves to be engineers. 38% of 79 respondents had engineering degrees, 61% had a computer science or other computing degree. The following respondent agreed that they considered themselves an engineer:

“Engineering methods are important to IT applications because the engineering knowledge is essential to understanding the fundamental relationships between the various data elements that are being stored, manipulated, operated-on, and communicated in IT programs, data bases, and networks.” (Manager of a computer-aided design firm in nuclear power plant design).

The following respondents did not believe themselves to be engineers:

“I generally feel that "Engineer" is a term which comes pre-loaded with its historical definition; i.e. mechanical, civil, etc.” (Contractor/Consultancy, energy and mining).

“I don't consider myself an engineer in the way in which engineers seem to define themselves. I note that some of our degrees fail [accreditation] because of lack of management courses, not lack of what I would think of as engineering.” (Professor, education).

“I do some of the time and am proud to be CEng/CITP but I disagreed because I deal primarily with the human and organisational aspects of IT-enabled business change and an engineering approach to this would be ineffective or destructive.” (Director and Principal Consultant, public sector company).

Comment on the findings: A number of respondents considered their IT roles to make them engineers. Some might have so identified because their educational background, but a number with computing degrees consider themselves engineers. However, one respondent specifically self-identified as a computer scientist, as opposed to an engineer.

It is interesting to note that at least one respondent who did not identify as an engineer ruled it out because their work had a human or organisational focus; while another respondent noted that management skills are important if a degree is to be accredited by an engineering institution.

Practitioners seem to recognise that engineering skills are relevant to all IT application domains but there is an added benefit from involving staff with an engineering background when the IT systems are part of a larger engineering application and the IT development team need to interact with professional engineers from other branches of engineering.

Why do some IT professionals not want to identify themselves as engineers? The following are possible reasons suggested by Andrew Ramsay of ECUK (the UK Engineering Council):

- Social: not wishing to be known as engineers, IT specialists may believe their status is higher and choose other definitions.
- Practice: an industry that has not traditionally regarded itself as a branch of engineering may not see the need for this label.
- Structural: different jobs within IT may be closer or further away from engineering (sales, information management), and a global definition would obscure this.
- Integrity: difficulty in forecasting the performance of complex IT systems may make them less characteristic of traditional engineering.

It is clear that IT presents some challenges not present in traditional areas of engineering, and that many professionals in the IT field are not, and do not need to be, engineers. However, there is also clear overlap between the sectors and the argument of this report is that in many IT areas engineering skills are crucially important. Chapter 4 describes the areas of IT which are closest to engineering and where engineering skills are of greatest importance.



3.3 IT and design tools

The mature engineering disciplines make use of analytical, mathematical and physical tools that are distillations of many decades of professional practice. The tools of a structural or mechanical engineer are a synthesis of experience and formal, scientific methods that underpin practice.

IT can, and should, make use of tools developed in this way. They are vital to bringing rigour and standardisation to the IT industry, as they are underpinned by well-founded and well-used methods. The table in Appendix C shows some of the mathematically based methods and tools that are used within the IT industry and the applications they relate to. Many other tools exist, of varying degrees of formality, which are appropriate to a range of industrial applications.

It is important that customers are able to establish the maturity and capabilities of software tools, to assess the appropriateness of the tool to the application and to have confidence in the dependability of software components. Industry bodies should work with professional societies to develop and promote an accreditation scheme that would provide customers with auditable evidence of the dependability, maturity and capability of software tools and components.



3.4 IT and mature engineering processes

Mature engineering processes are important to IT projects because they save effort, they reduce errors, they help to control risk, and they preserve the chain that connects the specification to the final product and to the evidence that the product is fit for purpose. There are a number of standards that are able to demonstrate that IT systems and processes are in accordance with mature processes.

The international standard ISO IEC 90003 explains how the generic standard for quality management systems, ISO 9001, can be applied to software and related services. The TickIT certification scheme (www.tickit.org) provides services to help software companies and departments to achieve ISO 9001 certification; this has recently been upgraded to TickIT Plus (www.tickitplus.org) which revises the services available and provides support for related standards ISO/IEC 20000 and ISO/IEC 27001.

In the US, the Software Engineering Institute at Carnegie Mellon University developed a Capability Maturity Model (CMM) at the request of the Department of Defense as a way of assessing the maturity of companies' software engineering processes and helping the companies to improve. The latest version of this, CMMI, was released in 2006 (www.sei.cmu.edu/cmmi/index.html).

ISO 9001 certification is broadly equivalent to Level 3 of the CMM and should be considered the minimum acceptable maturity for a professional software development organisation.

4. Engineering skills in IT projects – critical application areas and critical roles

IT professionals with engineering skills and understanding can contribute to the development of software-based IT systems in the following broad ways:

- Inspiring: explaining the technical possibilities – to present the opportunities for technical solutions that engineering presents.
- Gatekeeping: explaining what the technology *can't* deliver, to prevent impossible projects from being instigated.
- Facilitating: keeping an eye on the technical detail between the initial ideas and the requirements.
- Implementing: turning requirements into final products.

All IT systems can benefit from the application of engineering skills and methods. Rigour, professionalism and the used of experienced, competent experts is essential throughout the IT profession. However, some areas of IT stand in particular need of an engineering approach, and even the experience and knowledge of professionals with engineering qualifications. This chapter will set out the areas of IT systems development for which an engineering approach is necessary.

4.1 Application areas requiring engineering skills

The need for engineering skills depends on the complexity of a given project or type of IT development and on the degree to which it must be shown to be dependable before it is put into service. Engineering skills are particularly important in the areas of high technical complexity where there are well-established safety certification standards, such as avionics or automotive: failures in these kinds of system have the highest consequences for human safety. Major deep technical challenges also arise in meeting the non-functional requirements (such as high performance, resilience, and security) of non-certified but high consequence socio-technical systems supporting enterprises and major public-sector organisations. Therefore, IT systems engineers will also have an important role in eliciting and modelling the specifications of these more demanding IT systems. They will be crucially important in IT upgrades for major socio-technical systems such as those created for the NHS *Connecting for Health* project, though the particular challenges of these systems also require the input of experts outside of IT and engineering to precisely identify the complex requirements.

As a result, some areas of IT are likely to need a greater input from engineers than others. The list below gives examples of the kinds of IT systems that require engineering skills and the application of an engineering method.

Embedded systems – automotive

Engineering skills are crucial to embedded systems, where IT systems are an integral part of engineered products. The automotive sector is making increased use of embedded systems, with electronics and software becoming the dominant factor in automotive capability. Electronics have been estimated at 40% of the vehicle cost by 2010; this proportion is growing despite the difficulties in the sector, as it is the key means of providing product differentiation. Electronics and software has many uses in vehicles, e.g.:

- vehicle control, especially powertrain and safety, e.g. ABS
- comfort and convenience, e.g. voice operated controls
- infotainment, e.g. audio, games and navigation
- driver assistance, e.g. parking aids.

IT drives product innovation, and cycle times are being compressed in order for suppliers to stay ahead; thus the 60 month development times which were typical at the turn of the millennium are now dropping to 36 or even 24 months. This means that the automotive sector has to deliver high integrity (highly reliable, safe) systems in the sort of time-frames more associated with commercial developments. Automotive systems are at least matching aerospace systems in terms of innovation, and they are now driving many aspects of the embedded systems market, including publishing and requiring compliance with the MISRA-C guidelines for the use of the C language in critical systems. The UK has some highly successful specialist firms in this area, and these firms typically have a strong engineering ethos, so they could be used as sources of good practice for many in the embedded systems sector.

Engineering skills are important not only in the development of automotive systems, but their verification. The Vehicle Certification Agency type tests every model of car before it is judged to be roadworthy, and this includes the software embedded in the car. This also extends to information forensics for investigating accidents. Engineering skills are particularly useful here as they can combine an understanding of IT systems with an understanding of the engineered system into which they are integrated.



High consequence socio-technical systems – the NHS

Engineering skills are increasingly needed by the NHS's IT services, though they are not often employed directly. Engineering challenges for the NHS include putting modern IT infrastructure into old, listed buildings; installing reliable and secure wireless to assist clinicians as they move around the hospital; and making portable IT equipment secure. The NHS is also currently involved in the large *National Programme for IT* which involves converting existing patient records into electronic data, and digitizing new patient data close to the point of capture. This kind of huge IT-supported business change project needs, and receives, significant input from engineering because the user requirements are very complex, interacting and sometimes in conflict (such as the patient's

requirement that their records should be kept highly confidential, whilst making them accessible wherever they are treated and used for non-direct patient care activities such as billing, clinical audit, planning and management), and because errors in the system may put patients' health in jeopardy (for example, if two patients' records are transposed or records are corrupted or inaccessible for emergency care). NHS IT also has to operate in difficult environments, with the avoidance of patient infection a particular concern. The engineering skills needed for such large and critical systems include: programme and project management, requirements analysis, implementation, verification, testing and redesign of processes. The development of an IT system that must support the many and various functions of the NHS would benefit from the skills of engineers accustomed to working with large-scale socio-technical systems.

The NHS programme is just one of many major public-sector IT systems development programmes; others include systems for paying benefits, collecting taxes, border security, and so on. The procurers of such systems, most often in Government, must have access to the right skills in order to make these kinds of projects a success. This will involve having qualified staff to procure from the right sources to meet the right purposes; using appropriately qualified professionals to develop systems, which should mean using developers with chartered status; and accessing expertise to ensure that the work done by contractors is of the appropriate standard. Government would benefit from working with the professional institutions to take advantage of their expertise for the latter purpose.

High consequence commercial – gaming

The gaming sector represents an area of high-consequence commercial IT development. Where computer games may once have been written by enthusiasts in their spare time, the global computer games industry is now an entertainment business with revenues significantly bigger than the global movie industry. For example, the "massively multiplayer" online game *World of Warcraft* produced by Blizzard has 10m subscribers each paying \$15/month, generating cash revenues of \$1.8bn per year. Gamers can make real-world money by winning items in the game and selling them to other users. In this sense, *World of Warcraft* and other leading-edge online game services are high-consequence systems pushing lots of technical boundaries trying to keep the huge number of users happy. Each development project involves large teams of people whose work must fit together and who may need to make changes to the same software modules almost simultaneously, without introducing errors. Engineers have considerable experience of these difficulties and the disciplines of version control, configuration management and interface specifications have transferred very effectively to software engineering. The gaming industry has to be very rigorous in its development processes, whilst meeting tight delivery schedules (such as getting new games to retailers in time for Christmas). These are engineering challenges and the programmers working within the industry might rightly be considered to need engineering skills.

Another significant example of high-consequence commercial systems is the financial sector. Appendix D outlines the importance of engineering skills to a high-consequence IT project within the financial sector.

4.2 High consequence and high cost systems: licensing practitioners and recording failures

There are many areas of IT where there would be serious consequences if the systems are not developed on time or within an acceptable budget or if they fail once they have been put into service; it is essential that professionals working in these areas can demonstrate the expertise appropriate to their role. Employing Chartered Engineers or Chartered IT Professionals with relevant experience could go some way to guaranteeing appropriate leadership. However, because of the increasing complexity of IT systems and the increasingly serious consequences should they fail, it should be considered whether special qualifications, or licensing, should be introduced for some areas of IT systems engineering. Such licensing exists in some areas of engineering: for example, civil engineers working on dam inspection need a licence to practice because of the huge potential consequences should they fail to carry out their work successfully. It may be considered surprising that the technicians who install and maintain avionics systems in aircraft must be individually licensed, when the people who design and develop these systems have no such requirement.

Engineering institutions should consider the possibility of introducing licenses that are either voluntary or mandatory for particular classes of high consequence IT projects. Those licences could either show that the IT professional has the expertise relevant to that area, or could be required for anyone working in that area. The licence must be domain specific because issues in these areas require a total system understanding, involving such concerns as requirements, hardware design, user training and selection and design tool authentication. This should be carried out in collaboration with the Health and Safety Executive.

The Government procures some of the most high-consequence and costly IT systems, particularly in the high-consequence socio-technical domain. Government-procured IT systems have in the past met with significant delays and failures – such as the expensive and serially-delayed NHS Electronic Patient Record system and the Child Support Agency system. As discussed in the previous chapter, IT needs to learn from errors just as other areas of engineering do. It would be of great value to the IT industry in general if the Government could commission independent analyses of significant public-sector projects and publish them. Sharing information about the root causes of both successes and failures of Government IT projects would be of great benefit to the industry and could add to the learning of both IT professionals and those who make use of their skills and services.

There is potential to make great savings if engineering methods are used in IT systems development, and if lessons can be learnt from past failures. A paper published by Ryan Nelson, 'IT Project Management: Infamous Failures, Classic Mistakes, and Best Practices' outlines the impact of ten infamous IT project failures. Among them was 'Taurus', a paperless share settlement system, abandoned by the London Stock Exchange in 1993 at a cost of over £800 million, and 'Innovate' a digital network procured by McDonalds to track performance of equipment in its branches, which cost \$170million before being abandoned in 2003.³

³ R. Ryan Nelson, 'IT Project Management: Infamous Failures, Classic Mistakes and Best Practices' in MIS Quarterly Executive vol. 6 no. 2, June 2007

Engineering skills in IT: survey findings

69% of 84 respondents agreed that there were IT application areas where engineering methods were recognised and valued. The main areas were **safety critical systems, hardware, real time control systems, security and test engineering.**

70% of 80 respondents felt that there were IT applications where the importance of engineering methods was not recognised, but should be. These included:

data centre design, data engineering, system architecture, systems development, requirements engineering, testing and the design of tests, large scale software implementations, security, service delivery, finance industry, anything business-critical, research analysis management, change management, risk management, performance management, future-proofing, capacity. In the words of one respondent: "good engineering methods and discipline greatly improve the delivery and capability of most areas of IT".

60% of 82 respondents agreed that there were specific roles in IT projects that should be filled by a qualified engineer. These included: **project manager, development lead, governance, management, systems analyst, architect, user interface designer, designer, test manager, project manager, technical design authority, subject area experts, technical writers, implementers (installers), roles that involve sign-off of major IT projects, quality assurance, drafting specifications, lead programmers, technical applications such as used in engineering and some areas of manufacturing and process control.**

Comments on findings

The survey showed significant support for the view that some IT application areas, or specific roles in IT projects, need qualified engineers. The kinds of areas and roles are very varied, but the most frequently cited areas were critical systems, security and real time control systems, and the roles that most needed engineers were systems architects, testing and signing-off a major IT project. Some respondents suggested that engineers needed to be involved across the whole spectrum of IT application areas and roles – one commented "I would prefer to see most staff to be of an engineering mindset to ensure the 'right' mindset". However, there are stereotypes abounding which may make it hard for some to argue this case: "Difficult one. I'd like to say 'most of them' but since 'engineer' in the world of IT invariably equates to 'the bloke that comes in to fix the photocopier', I'm not sure any more."

5. Education, training and research for success in IT

5.1 Education and training

Universities have long taught the core of the IT discipline, and this remains relevant to systems design and development today. However there are three major ways in which education and training in software development needs to develop to support industry:

- Undergraduate education (initial professional development) needs to be adapted to reflect better the realities of large scale software development, e.g. the complexities of embedded systems, the dependability issues for enterprise systems, the fact that most developments are “brown field”, i.e. adapting existing systems rather than developing new ones, and so on.
- It should be recognised that a joint or combined three-year degree, such as ‘IT with Management’ cannot provide the depth of education in software and systems engineering that is needed by a professional software engineer, and that degrees that focus on specific niche applications, such as ‘Web Application Design’, are unlikely to qualify graduates to play a leading part in systems outside their specific niche. Such degrees can be very valuable for staff who intend to work in areas that the degrees target but they are unlikely to provide sufficient educational formation for a professional engineer.
- Continuing professional development needs to be strengthened to reflect the increasing capability and relevance of the research results developed in universities and other research centres to ensure that competent professionals are able to deploy the most effective tools and techniques on their problems.

Of course, there is some good practice in this area – but much more needs to be done, and this requires effective collaboration between universities and industry to ensure that education and training is effective. The Sector Skills Council for Business and Information Technology, e-Skills UK, is already interacting with Universities, e.g. to stimulate relevant undergraduate education, and they are in the position to act as an effective change agent.

A further key issue is the attractiveness of the subject to prospective undergraduates; again collaboration between Universities and industry/commerce is relevant as companies can help in marketing of appropriate courses. Also, one of the long-standing problems in recruiting into computing has been the curricula taught in secondary schools. School teaching of ICT focuses on the use of IT rather than its development (see section 7.1) and therefore fails to convey the importance of engineering skills and understanding to IT development. A report shortly to be published by The Royal Academy of Engineering on the future of UK ICT competitiveness expands on this point. Recently, there has been an acceptance that it is necessary to re-consider the content and emphasis of the teaching of IT in schools where, again, e-Skills UK are taking the lead (with assistance from CPHC – Council of Professors and Heads of Computing).

5.2 Research in industry and academia

The importance of collaboration extends into the research arena. Whilst there is always merit in basic research, from which real innovations can arise, there is also a need to nurture the transfer of research into practice. This involves a two-way flow between industry and researchers, especially in Universities – to provide the research community with a good understanding of real-world problems and to adapt the research results for practical use.

For example, most industrial software developments are “brown field” adapting and integrating existing systems to provide business benefit. Relatively little academic work focuses on these systems – and there are some very difficult issues in analysis of legacy systems, change management, systems integration, and regression testing⁴, for example, which would benefit from greater academic attention.

Also, there are many examples of promising research results which have not had the impact they deserved on industrial practice. Arguably this has been the case with formal methods although, recently, there is greater take-up of these techniques, see the table of tools in Appendix C. Generally the difficulty is one of the perceived risk of using the techniques before they are fully mature – put another way, few are willing to be a first adopter. Funds for maturation of tools, for experimental validation of tools, for collecting and analysing metrics, for example, would be instrumental in overcoming such problems.

Again there are examples of good practice in this area (excellent work is being done, for example, in the UK Computing Research Committee / EPSRC Grand Challenge project on Dependable Systems Evolution), but more needs to be done to foster collaboration – and this is an area where effective coordination of programmes between the EPSRC and TSB should yield benefits. However care should be taken not to stifle basic research – innovations often take 15-20 years from initial idea to industrial application, so the lack of an immediate application should not be a reason for failing to fund high-quality basic research.



⁴ Testing for regression, where changes to a system can cause failures in parts that previously functioned without problems, or cause problems that had been addressed to recur.

Outside of the university system, management schools and training providers also need to strengthen their continuing professional development programmes to reflect the nature of demanding IT projects. Training providers must make use of the increasing capability and relevance of the research results being developed in academia and industry in order to ensure that they deliver up to date insight. It is important that training providers remain abreast of current research and technology, and ensure that training delivers relevant technical content alongside skills for dealing with business change.

Education and skills for IT success: survey findings

Of 72 correspondents, 33% agreed with the statement “my organisation actively seeks to recruit individuals with an engineering degree”, 29% did not agree, and 29% neither agreed nor disagreed. In comparison 44% agreed with the same statement relating to a computer science degree, with 13% disagreeing and 36% neutral. 72% agreed that experience was the determining factor, irrespective of degree. Respondents judged experience by CV rather than any more formal tools such as competencies according to a skills framework. However, one respondent reported that they looked for professional certification such as CCNA (Cisco Certified Network Associate) when recruiting.

Although the particular skills and areas of experience that respondents thought most important depended very much on the particular role that was recruited for, respondents gave the following responses for areas of experience they considered most important in recruiting: **project management; interpersonal/client-facing skills; coding; testing; time management; design; problem finding and solving; project ownership; effective presentation; whole system lifecycle: hardware & software knowledge; general commercial experience; requirements engineering; systems development; technical support; lifecycles; business analysis; implementation ("getting hands dirty"); risk management; interpretation of business environment and alignment to IT capabilities; planning / estimating; value management; operational support; diagnosis of hard problems; technical knowledge (language, database, networking; communication skills.**

Comments on findings:

Although there was not strong agreement from respondents that their organisations actively sought out candidates with engineering degrees, some of the areas of experience considered most important were common with other areas of engineering. These included project management, testing, design, requirements engineering, risk management. These also included the kinds of roles which some respondents previously suggested should be filled by engineers. However, some skills were noted that were perhaps specific to IT, or which particularly reflected the demands of IT development: “Ability to visualise that which is invisible, see the inter-relationships between all the system elements and be able to clearly communicate and explain them. The most difficult part of any project is maintaining a similar mental picture of what's being built amongst a group of people, then being able to discuss it, refine it, take sensible decisions, and deliver it”

6. Professionalism and ethics in IT

6.1 Why is chartered status important for IT professionals?

Chartered status is potentially of great value to IT professionals as it can demonstrate that they have the education, experience and competence to work on complex and high consequence IT systems. As IT systems become increasingly integral to critical infrastructure, there will be a duty on those who procure such systems to ensure that they are employing professionals who can meet the high standards needed and who can take responsibility for such high-consequence systems.

Why would we want a chartership scheme? It sets a norm, and is a transferable qualification. Chartered status sets a benchmark to judge training by. Chartered status is a passport between different jobs and companies.

Peter Brook, QinetiQ

The survey results show that employers currently consider experience to be the most important factor in employing IT professionals, regardless of qualification. It is therefore important to be able to judge the level of expertise of IT professionals working in different companies and on different projects. Chartered status can be of value in demonstrating a certain level of experience and competence, wherever that was acquired. In fact, chartered status is particularly valuable as it is a demonstration of competence rather than simply experience – showing that an individual has acquired appropriate skills to a high level from their professional experience.

There is therefore a motivation for IT professionals to become chartered. However, there is also a duty for those who use IT companies to ensure that the professionals they work with really do meet the necessary standards. Selecting those whose lead staff are chartered can help them to meet that obligation.

Currently two charter schemes are relevant to the IT Professional; the Chartered Engineer (CEng) under the auspices of ECUK and the Chartered IT Professional (CITP) being developed by the BCS.

6.2 Qualifying as a Chartered Engineer

In the UK Professional Engineers who have been chartered by an authorised engineering institution are referred to as 'Chartered Engineers'. According to the UK Engineering Council (ECUK), Chartered Engineers are:

Characterised by their ability to develop appropriate solutions to engineering problems, using new or existing technologies, through innovation, creativity and change. They might develop and apply new technologies, promote advanced designs and design methods, introduce new and more efficient production techniques, marketing and construction concepts, or pioneer new engineering services and management methods. Chartered Engineers are variously engaged in technical and commercial leadership and possess effective interpersonal skills.⁵

To qualify as a Chartered Engineer (and hence use the postnominal letters 'CEng'), an engineer must exhibit certain competencies specified by ECUK. The following is a summary of those competencies (details are in appendix B):

- a) Use a combination of general and specialist engineering knowledge and understanding to optimise the application of existing and emerging technology.
- b) Apply appropriate theoretical and practical methods to the analysis and solution of engineering problems.
- c) Provide technical and commercial leadership.

⁵ See <http://www.engc.org.uk/registration/CEng.aspx>

- d) Demonstrate effective interpersonal skills.
- e) Demonstrate a personal commitment to professional standards, recognising obligations to society, the profession and the environment

The entry point for Chartered Engineers is an accredited Bachelors degree in a relevant engineering or technology discipline, plus either an appropriate Masters degree or experience and training to an equivalent level. For some areas of IT, the specialist skills and education of an engineer will be needed. These were outlined in chapter 4, and they include the highest consequence IT systems. Engineers are particularly suited such areas as their mathematical background and the formal methods they learn allow them to develop systems with rigour using scientifically-founded methods.

The value of becoming a CEng is that the qualification is a badge of quality recognised by many employers globally. Chartered status is a mark of recognition that a professional has built up the requisite experience to work on a range of projects at a high level of authority. Being a member of a professional engineering institution also brings with it commitment to a professional ethical code which commits the engineer to a high level of professionalism in their practice.

6.3 Qualifying as a Chartered IT Professional (CITP)

CITP status is awarded by the BCS, and like the CEng, CITP status brings with it the commitment to ethical standards and to continuing professional development. However the distinguishing feature of entry to the CITP qualification is its strong emphasis on competence based on the SFIA (Skills Framework for the Information Age) model. Further developments of the CITP qualification are expected to require demonstration that an IT professional has achieved a defined level of competence in a wide range of areas alongside high levels of competence in a smaller subset, representing their specialisation. Whilst CITP recognises professionals in a wide range of specialisms within the broad field of information technology systems management, it is envisaged that one of those specialisms will be rigorous application of systems design in the development of IT systems. This would be awarded to those professionals with knowledge of the appropriate underlying science and engineering methods for the development of IT systems. Professionals who achieve this CITP (engineering) qualification would therefore be considered qualified to lead or manage major IT projects.



Chartered status is a good thing. Chartered status is valued in many professions, but it does not have the same profile in engineering and IT. This makes it difficult for the small employer looking for IT staff.

John Suffolk, Government Chief Information Officer

6.4 The need for professional development

Continuing professional development (CPD) is the process by which chartered professionals maintain their knowledge and understanding, and remain equipped to apply professional judgement. Engineering depends on experience and this is gained throughout an engineer's career. Development of professional experience can happen at the same time as formal education, for example through an industrial placement during a higher education course, or alongside part-time study.

Ongoing professional development is of particular importance within IT, as this is an area where technologies and the environments in which they are applied are constantly changing. Therefore, institutions that offer chartered status to IT professionals should give particular support to members in maintaining their professional development. The BCS and IET are both committed to attracting IT professionals to their membership, and providing a clear CPD route for those professionals, allowing them to maintain up-to-date knowledge and skills. Given the scale of new knowledge being generated, it is envisaged that IT professionals will focus their CPD on the areas of appropriate to their domain specialism.

As argued in chapter 4, chartered status (either CEng or CITP) may be augmented by special registration or a licensing procedure for engineers working on projects with particularly serious consequences. Professional institutions should actively encourage IT professionals to achieve chartered status.

6.5 Ethics in IT

As a Chartered Engineer or a Chartered IT Professional a professional is bound by the code of ethical conduct of his or her institution. These codes of ethics ensure a high standard of professionalism⁶. The Royal Academy of Engineering and ECUK's statement of ethical principles for the engineering profession is intended to capture the essence of all the institutions' ethical codes, and has been adopted by the majority of institutions. The statement places a number of responsibilities on engineers which are very relevant to the concerns of this report. The statement of ethical principles (included in full at appendix E) is grouped under four headings: Accuracy and Rigour; Honesty and Integrity; Respect for Life, Law and the Public Good; Responsible Leadership: Listening and Learning. All of these are relevant to the IT professional, but the challenges of creating novel and demanding IT systems for clients bring a particular need for commitment to the values of accuracy and rigour.

The statement of ethical principles lists the following duties, amongst others, under the heading accuracy and rigour:

Professional Engineers have a duty to ensure that they acquire and use wisely and faithfully the knowledge that is relevant to the engineering skills needed in their work in the service of others. They should:

- always act with care and competence;
- perform services only in areas of current competence;
- keep their knowledge and skills up to date and assist the development of engineering knowledge and skills in others;
- identify and evaluate and, where possible, quantify risks.

⁶ The BCS code of conduct is online here: www.bcs.org/server.php?show=nav.6030
The IET rules of conduct are online here:
www.theiet.org/about/governance/rules-conduct/index.cfm

These principles are particularly important to IT professionals who are working in a fast developing technological environment. They entail a duty on the IT professional to keep their knowledge up to date. Without an understanding of up to date technologies, tools and methods, systems developers will be badly serving their clients who pay for the best available solution to meet their needs.



There are a number of ways that the profession can support its members in adhering to these principles. This report recommends that IT professionals achieve chartered status, that institutions support them in keeping their professional knowledge up to date, and that, where appropriate, special registration demonstrating competence to practice is required for IT professionals. We also recommend that the CEng and CIPD require CPD and revalidation to reflect current skills and competencies. By these processes, it can be ensured that IT professionals can demonstrate that they are working within their competencies, using up to date knowledge.

Professionals have an ethical responsibility to pass on learning to their peers, especially when that pertains to failures that could be avoided in the future. We recommend that information about the procurement of major IT systems is shared to assist in this professional learning. In particular, the Government as the procurer of many large-scale systems, should report openly and independently on the lessons learned from its IT successes and failures alike. These measures will help the engineering and the IT profession to meet these ethical responsibilities, ensuring that IT is a trusted and ethical profession.

7. Understanding the barriers – how engineering approaches can be implemented in IT

A number of misapprehensions about IT systems engineering have become apparent during our study. The main ones are addressed below.

7.1 Confusion between developing and using IT

Much of the debate about IT skills reveals a confusion between the skills required to make effective use of IT within an organisation and the skills needed to develop new IT systems. The skills that are necessary to use IT effectively in any organisation range from basic knowledge of how to use a word processor, spreadsheet or database, to understanding the potential that software packages and networking have to support and transform business processes. IT system developers need these skills too, but they also need deep technical understanding of computing and communications technology, knowledge of computing science and engineering methods and the creativity to combine their skills and knowledge to create a cost-effective and dependable solution to a set of complex requirements. So, whilst managing the use and implementation of IT systems within an organisation may not always require engineering skills, IT development does require engineering skills, knowledge and methods.

The visual or intuitive element is missing from IT. It sounds silly to design an aircraft with wings a mile wide, but does it sound silly to try to build a computer system for the whole of the NHS in one step?

Peter Brook, QinetiQ

7.2 What is conceivable is not always possible

As argued in *The Challenges of Complex IT Projects*, one of the obstacles to engineering methods being successfully implemented in IT is the fact that IT systems intended for novel or challenging purposes can be conceivable to a client, and seem easy to create, but may in fact be impossible to develop successfully. Thus systems that seem a good idea are planned and contracts may be placed even though they may be impossible to develop successfully.

One might be able to imagine, for example, designing a car that could always safely absorb impacts no matter how severe they were and such a product would be highly desirable. However, most people are aware that this is not likely ever to be possible, however sophisticated car design becomes. Similarly, it may seem desirable to have one unified electronic patient record system for the whole of the NHS, available everywhere and yet secure, but in this case, the sheer difficulty of producing such a system, and its possible disadvantages, are not so obvious. It may not be clear that it is extremely difficult to build one system to meet a complex range of social needs, building on a range of existing IT systems which work in quite different ways. Too often, large-scale IT systems are planned and procured which are almost impossible to deliver.

The particular challenge for IT is that of *externalising design* in order to judge whether an IT system could be developed successfully. There is also the difficulty that, for those outside the IT profession, there is a lack of intuitive sense of the complexity of IT systems, which is not the case for other systems such as those in traditional areas of engineering. A professional approach to IT should involve establishing with the client just what the barriers are in creating a system and should therefore involve working towards a realistic solution. A key step in this process is to explore and model the requirements of all the stakeholders, so that contradictions can be identified and resolved and challenges identified before development contracts are agreed. The drive to secure lucrative contracts should not lead developers to taking on projects that they cannot be sure they can deliver.

7.3 Testing is not enough

When a system fails in service, the observation is often made that “it wasn’t tested enough”. This shows a serious misunderstanding of the nature of software-based systems because, while testing of an IT system is important, it cannot assure the safety, security or reliability of a system. The discrete nature of digital systems means that a successful test tells you little or nothing about how the system would behave if the inputs were even slightly different, or presented in a different order. The complexity of a large IT system means that the number of tests that would be required to provide adequate evidence for even modest requirements would take an impractical amount of time to create and perform. The only way in which strong evidence for dependability can be created is to ensure that the requirements were clearly expressed and analysed for completeness and consistency, that the boundaries between what was known and what was assumed are clear and clearly stated, and then verifying that the developed system completely implements the requirements. That is, the system has to be built from secure foundations in intrinsically secure steps, and the security of those foundations and steps provides the assurance that a system will perform as expected. Testing should be used to validate the assumptions and as a check to give confidence that analyses of the system were accurate. Any failure during testing should be analysed to understand the root causes – how the error came to be introduced and why the development processes failed to detect the error earlier – so that other errors of this class are found too and so that the processes can be improved to avoid such errors in future.

7.4 Errors should no longer be regarded as inevitable

It is often argued that software, by its nature, will always contain many errors and that manufacturers and importers of software products should not be subject to the same product liability and consumer protection regimes that apply to other products. Whilst there was some merit in this argument in the early days of the software industry, it is now possible to develop software products with very few errors and to do so cost-effectively, using science based software engineering methods. A high level of errors is therefore no longer inevitable and should not be tolerated in software-based systems to any greater extent than they are in other engineering artefacts. See www.adacore.com/tokeneer for an example of a project attempting to create an error-free IT system. Although this is a modest system in comparison to those we are concerned with here, it is an important step towards developing complex software-based systems without errors.

7.5 Science based methods are not impractical

Appendix C gives some examples of software engineering methods that are founded in rigorous computer science, supported by industrial-strength tools, and in routine use in a variety of industrial applications. There are many more examples that could be cited, and the evidence clearly shows that mathematically formal software engineering methods have matured to the point where they are practical and cost effective for routine use. IT professionals should be familiar with the rigorous methods that are available and appropriate for their work and should adopt them where this will be of benefit, alongside other established means of ensuring software reliability such as fault tolerance and hazard analysis.

8. Concluding remarks

There is no doubt that information systems are important to the national economy and governance. Many of those systems must be classified as of high consequence since they critically impact on issues such as safety, security or finance. Accordingly the development of such systems places great responsibility on those undertaking their construction and demands the highest levels of professionalism and expertise.

Central to this report is the question of the value of an engineering approach in the development of IT systems. The evidence we have evinced strongly supports the value of an engineering approach to IT, though whether the practitioners regard themselves as engineers is a matter of individual opinion and is not of crucial consequence. IT is, of course, a broad profession and many areas of IT do not require individuals with engineering backgrounds or qualifications. However, within IT system development, and leaving aside the straightforward applications developed using office support suites or web authorship tools, development of IT systems benefits from the application of proven engineering processes and increasingly from the application of mathematically based design tools. The development of staff with these capabilities was identified as a major challenge for the educational community.

The importance of IT to critical infrastructure and services requires professionalism on the part of practitioners. Professionalism requires the adherence to a strong ethical code and a commitment to lifelong learning as demonstrated by the achieving chartered status, either Chartered Engineer or Chartered IT Professional. Senior management stressed to the study group the potential value of chartered status in employing staff. Indeed, some advocated the creation of application specialities within the chartered status which would be mandatory for certain very high consequence systems.

In sum, it is clear that IT is of crucial importance to our daily lives. As with traditional areas of engineering, engineering method and skills and a commitment to professional standards are essential to maintaining our IT infrastructure and hence our essential services.



9. Appendices

Appendix A: Methods

This report was carried out by conducting a survey of membership groups within the IET and the BCS; interviewing representatives in several key sectors of industry commerce and government; and working group discussion.

Terms of reference

The terms of reference of the working group are as follows:

1. to identify the roles within an IT project where the utilisation of an engineering approach would substantially benefit the project
2. to consider how the currently conceived CEng would enable qualified engineers to be effective in the applications and roles identified in TOR 1
3. to consider whether the currently available academic and industrial education provides the requisite core body of knowledge at the Masters Level as mandated by the CEng qualification and recommend how these offerings might be continually developed
4. to advise on areas of research which need to be commissioned (or brought to a state of utility) to enhance the core body of knowledge
5. to consider how to stimulate uptake of the findings of the study by the management in organisations which commission and/or undertake the development of IT systems

Working group members

Dr Allan Fox FREng FBCS FIET (Chair)

formerly Nortel Networks

Prof Dave Cliff FBCS

University of Bristol

Sir Tony Hoare FREng FRS FBCS

Microsoft Research

Mr Max Fordham OBE FREng

Max Fordham Consulting Engineers

Mr Nick Masterson-Jones

Vocalink

Prof John McDermid FREng FBCS FIET FRAeS FSaRS

University of York

Ms Michelle Richmond FIET

IET

Dr Mike Rodd FBCS FIET

Formerly BCS

Dr Martyn Thomas CBE FREng FBCS FIET

Martyn Thomas Associates Ltd

With thanks to:

Melissa Bailey FBCS, Mike Glover FREng, Robert Berry FBCS,
Jim Norton FBCS FIET

Appendix B: Engineering competences

A list of skills needed by the professional engineers as given in the UK Engineering Council's (E^{CUK}) UK-SPEC (UK Standard for Professional Engineering Competence):

- A** Use a combination of general and specialist engineering knowledge and understanding to optimise the application of existing and emerging technology
- Identify the limits of own personal knowledge and skills
 - Strive to extend own technological capability
 - Broaden and deepen own knowledge base through research and experimentation
 - Establish users' needs
 - Assess marketing needs and contribute to marketing strategies
 - Identify constraints and exploit opportunities for the development and transfer of technology within own chosen field
 - Promote new applications when appropriate
 - Secure the necessary intellectual property rights
 - Develop and evaluate continuous improvement systems
- B** Apply appropriate theoretical and practical methods to the analysis and solution of engineering problems
- Explore for new opportunities, assess viability
 - Review potential for enhancing products, processes, systems and services
 - Identify appropriate research methodologies
 - Assemble necessary resources
 - Carry out necessary tests
 - Analyse and evaluate data
 - Present/agree design recommendations
 - Undertake design
 - Ensure appropriate practical outcome, identifying cost, quality, safety, reliability, appearance, fitness for purpose, environmental impact
 - Determine criteria for evaluating the design
 - Evaluate outcome against specification
 - Learn from feedback

- C** Accept and exercise personal responsibility.
- Work reliably and effectively without close supervision, to the appropriate codes of practice
 - Accept responsibility for work of self and others
 - Accept, allocate and supervise technical and other tasks.
- D** Use effective communication and interpersonal skills.
- Use oral, written and electronic methods for the communication in English of technical and other information
 - Work effectively with colleagues, clients, suppliers and the public.
- E** Make a personal commitment to an appropriate code of professional conduct, recognising obligations to society, the profession and the environment.
- Comply with the Code of Conduct of their Licensed Institution or Professional Affiliate
 - Manage and apply safe systems of work
 - Undertake engineering work in a way that contributes to sustainable development

Carry out continuing professional development, including opportunities for this offered by their Institution, to ensure competence in areas and at the level of future intended practice.

Appendix C: Software design tools

The information in this table was extracted from *Verification and Formal Methods: Practice and Experience*. (J Woodcock et al, ACM Surveys, to be published).

Name of tool	Reference	Industrial application
ASTREE	Blanchet, B., Cousot, P., Cousot, R., Feret, J., Mauborgne, L., Min'ée, A., Monniaux, D., and Rival, X. 2003. A static analyzer for large safety-critical software. In PLDI. 196–207.	Examination of C code for potential run-time errors. Used on Airbus A340/A380 software.
B	Abrial, J.-R. 2007. Formal methods: Theory becoming practice. JUCS 13, 5, 619–628.	Formal specification language. Widely used for applications including and commerce, security and transportation.
Eiffel	Meyer, B. 1992. Eiffel: The Language. Prentice Hall.	Programming language supporting design-by-contract. Used in industrial projects including banking and finance, defence and aerospace, health care, telecommunications and game programming.
KIV	Haneberg, D., Schellhorn, G., Grandy, H., and Reif, W. 2008. Verification of Mondex electronic purses with KIV: from transactions to a security protocol. Formal Asp. Comput. 20, 1, 41–59.	Theorem prover: used to prove properties of protocols for the Mondex smart cards.
PREfix/PREfast	Bush, W. R., Pincus, J. D., and Sielaff, D. J. 2000. A static analyzer for finding dynamic programming errors. Softw., Pract. Exper. 30, 7, 775–802.	Microsoft: analysis of large programs in C, C++ for null pointer references and other errors.
SCADE	Esterel Technologies. 2003. SCADE Language Reference Manual.	Programming system based on the synchronous data-flow programming language, Lustre. Many applications in the transportation, energy, and defence domains.
SLAM	Ball, T. and Rajamani, S. K. 2002. The slam project: debugging system software via static analysis. In POPL. 1–3.	Microsoft and users of the Windows Development Kit: analysis of correct API usage in device drivers
SPARK	Barnes, J. 1997. High Integrity Ada: The SPARK Approach. Addison-Wesley.	Programming language, analysis and proof toolset. Used widely for applications including and commerce, security and transportation.

SPIN	Holzmann, G. J. 2004. The SPIN Model Checker: Primer and Reference Manual. Addison-Wesley.	Analysis of distributed systems for deadlocks, race conditions and other errors. Used on Deep Space 1, Cassini, Mars Exploration Rovers, and Deep Impact space missions.
VDM	Jones, C. B. 1990. Systematic Software Development Using VDM, 2nd ed. Prentice-Hall International.	Formal specification language. Used by Praxis for air traffic management system CDIS.
VDMtools	Fitzgerald, J., Larsen, P. G., and Sahara, S. 2008. VDMTools: Advances in support for formal modeling in VDM. Sigplan Notices 43, 2 (February), 3–11.	Support toolset for VDM++. Used for the mobile FeliCa IC chip firmware.
Z	Spivey, J. M. 1989. The Z Notation: a Reference Manual. International Series in Computer Science. Prentice Hall.	Formal specification language. Widely used for applications including commerce, security and transportation.

Appendix D: Case study – the financial sector

Nick Masterson-Jones, IT director at VocaLink, explained that an engineering approach was important to VocaLink because it was part of the UK critical infrastructure, handling over €4 trillion worth of transactions in 2006.

In adopting a new system, they required a system that had no single point of failure, which was capable of ultra high volumes of transactions with absolute integrity of data. An engineering approach meant certainty of technical outcome (having proof that a system will work); certainty of schedule (with precise deadlines); clear accountability and certainty of quality.

Requirements:

- No single point of failure anywhere
- Certainty of delivery within time window
- Absolute data integrity
- Capable of transacting ultra-high volumes
- Use of modern technology stack – Sun Solaris, Oracle, BEA Weblogic
- Service Oriented Architecture for re-use and flexibility

The Engineering Approach:

- **Certainty of technical outcome (will it work?)**
 - Early prototyping and Proofs of Concept to remove risk
- **Certainty of schedule**
 - Very tight programme management enforced
 - Culture of transparency enforced
 - “Do the hard stuff first” focus
 - Obsessive focus on delivery on early key milestones
- **Clear accountability**
 - All role profiles clearly articulated and written down
 - No black box development
 - Contingency plans always in place for delivery
- **Certainty of quality**
 - Architects executing high level design and frameworks
 - Development conducted in a “software factory”
 - Testers in a separate line function from development
 - Specialist testers and tools for separate test phases
- **Certainty of delivery outcome**
 - “Safety claim” approach used for go-live assessments
 - Extensive parallel run (2bn payments processed before go-live)
 - Quantitative modelling approach used for risk assessments on programmes
 - Independent assurance team checking progress through “quality gates”

Appendix E: Statement of Ethical Principles

The Statement of Ethical Principles is intended to unite the various codes of conduct and codes of ethics of the engineering institutions, including the codes of the BCS and the IET, drawing out the values that underpin the codes specific to each institution. It does not replace these specific codes, but seeks to set out the shared values and aims of the various branches of engineering and of IT.

The Royal Academy of Engineering, in collaboration with Engineering Council (UK) and a number of the leading professional engineering institutions, has created a Statement of Ethical Principles to which it believes all professional engineers and related bodies should subscribe.

Professional Engineers work to enhance the welfare, health and safety of all whilst paying due regard to the environment and the sustainability of resources. They have made personal and professional commitments to enhance the wellbeing of society through the exploitation of knowledge and the management of creative teams.

This Statement of Ethical Principles sets a standard to which members of the engineering profession should aspire in their working habits and relationships. The Statement is fully compatible with the principles in the UK Government Chief Scientific Adviser's Universal Ethical Code for Scientists*, with an emphasis on matters of particular relevance to engineers. The values on which it is based should apply in every situation in which professional engineers exercise their judgement.

There are four fundamental principles that should guide an engineer in achieving the high ideals of professional life. These express the beliefs and values of the profession and are amplified below.

* <http://www.dius.gov.uk/~media/publications/file41318>

Accuracy and rigour

Professional Engineers have a duty to ensure that they acquire and use wisely and faithfully the knowledge that is relevant to the engineering skills needed in their work in the service of others.

They should:

- always act with care and competence.
- perform services only in areas of current competence.
- keep their knowledge and skills up to date and assist the development of engineering knowledge and skills in others.
- not knowingly mislead or allow others to be misled about engineering matters.
- present and review engineering evidence, theory and interpretation honestly, accurately and without bias.
- identify and evaluate and, where possible, quantify risks.

Honesty and integrity

Professional Engineers should adopt the highest standards of professional conduct, openness, fairness and honesty. They should:

- be alert to the ways in which their work might affect others and duly respect the rights and reputations of other parties.
- avoid deceptive acts, take steps to prevent corrupt practices or professional misconduct, and declare conflicts of interest.
- reject bribery or improper influence.
- act for each employer or client in a reliable and trustworthy manner.

Respect for life, law and the public good

Professional Engineers should give due weight to all relevant law, facts and published guidance, and the wider public interest. They should:

- ensure that all work is lawful and justified.
- minimise and justify any adverse effect on society or on the natural environment for their own and succeeding generations.
- take due account of the limited availability of natural and human resources.
- hold paramount the health and safety of others.
- act honourably, responsibly and lawfully and uphold the reputation, standing and dignity of the profession.

Responsible leadership: listening and informing

Professional Engineers should aspire to high standards of leadership in the exploitation and management of technology. They hold a privileged and trusted position in society, and are expected to demonstrate that they are seeking to serve wider society and to be sensitive to public concerns.

They should:

- be aware of the issues that engineering and technology raise for society, and listen to the aspirations and concerns of others.
- actively promote public awareness and understanding of the impact and benefits of engineering achievements.
- be objective and truthful in any statement made in their professional capacity.

Appendix F: Survey – The Role of Engineers and Engineering in IT

The survey quoted throughout this report was distributed to members of specialist panels of the BCS and the IET, and was also distributed amongst the BCS's Young Professionals Group. The survey was run online, and the numbers of respondents quoted in the report indicates the number of respondents that answered each question (not all respondents answered all questions). The full list of questions and summary of responses are given below.

1. Which description most closely fits your role?

Company Manager or Director: 42.4% (36)

HR: 0.0% (0)

In-House IT staff: 57.6% (49)

Total responses: 85

2. Qualified engineers bring added value to our IT department.

Agree: 81.4% (70)

Neither Agree nor Disagree: 15.1% (13)

Disagree: 3.5% (3)

Total responses: 86

3. I consider myself to be an engineer.

Agree: 78.8% (67)

Neither Agree nor Disagree: 10.6% (9)

Disagree: 10.6% (9)

Total responses: 85

4. I have the following qualifications:

Engineering degree or other engineering qualification: 38.0% (30)

Computer Science degree or other computing qualification: 60.8% (48)

Degree in other science or humanities subject: 24.1% (19)

Other academic or vocational qualifications: (7)

Total responses: 79

5. There are IT application areas where the importance of engineering methods are really recognised and valued

Agree: 69.0% (58)

Neither Agree nor Disagree: 15.5% (13)

Disagree: 15.5% (13)

Total responses: 84

6. There are IT application areas where the importance of engineering methods is not recognised, but should be.

Agree: 70.0% (56)
Neither Agree nor Disagree: 22.5% (18)
Disagree: 7.5% (6)
Total responses: 80

7. Specific roles exist in IT projects where that role should be filled by a qualified engineer.

Agree: 59.8% (49)
Neither Agree nor Disagree: 31.7% (26)
Disagree: 8.5% (7)
Total responses: 82

8. Having Chartered Engineering Status (CEng) is valued in my organisation.

Agree: 31.5% (23)
Neither Agree nor Disagree: 32.9% (24)
Disagree: 26.0% (19)
Don't know: 9.6% (7)
Total responses: 73

9. Having Chartered IT professional status (CITP) is valued in my organisation.

Agree: 19.2% (14)
Neither Agree nor Disagree: 41.1% (30)
Disagree: 30.1% (22)
Don't know: 9.6% (7)
Total responses: 73

10. My organisation actively seeks to recruit individuals with an engineering degree.

Agree: 33.3% (24)
Neither Agree nor Disagree: 29.2% (21)
Disagree: 29.2% (21)
Don't know: 8.3% (6)
Total responses: 72

11. My organisation actively seeks to recruit individuals with a Computer Science degree.

Agree: 44.4% (32)
Neither Agree nor Disagree: 36.1% (26)
Disagree: 12.5% (9)
Don't know: 6.9% (5)
Total responses: 72

12. My organisation actively seeks to recruit individuals with Chartered Engineer (CEng) status.

Agree: 15.5% (11)
Neither Agree nor Disagree: 35.2% (25)
Disagree: 36.6% (26)
Don't know: 12.7% (9)
Total responses: 71

13. My organisation actively seeks to recruit individuals with Chartered IT Professional (CITP) status.

Agree: 9.9% (7)
Neither Agree nor Disagree: 35.2% (25)
Disagree: 43.7% (31)
Don't know: 11.3% (8)
Total responses: 71

14. There is a core body of knowledge that any IT professional should be expected to demonstrate.

Agree: 85.9% (61)
Neither Agree nor Disagree: 12.7% (9)
Disagree: 1.4% (1)
Total responses: 71

15. Experience is the dominant factor when employing an IT professional, regardless of degree/chartered status.

Agree: 72.2% (52)
Neither Agree nor Disagree: 15.3% (11)
Disagree: 11.1% (8)
Don't know: 1.4% (1)
Total responses: 72

**16. In what areas is it most important for candidates to have experience?
Eg, design, project management, risk assessment?**

See box, section 5.2

17. Which description most closely fits your organisation?

Predominantly IT company: 29.6% (21)

Company with an IT department: 38.0% (27)

Company dependant on procured IT systems: 4.2% (3)

Contractor/Consultancy: 28.2% (20)

Total responses: 71

18. Which sector most closely describes your company?

Aerospace and Defence: 5.3% (3)

Autos: 1.8% (1)

Basic Industries: 0.0% (0)

Consumer Industries: 3.5% (2)

Transport: 0.0% (0)

Drugs and Healthcare: 3.5% (2)

Energy and Mining: 1.8% (1)

Financial Services: 10.5% (6)

IT: 43.9% (25)

Public Sector: 14.0% (8)

Media and Internet: 5.3% (3)

Property: 0.0% (0)

Retail and Leisure: 1.8% (1)

Telecoms: 8.8% (5)

Other: (17)

Total responses: 57

19. Size of your organisation.

0-10: 9.9% (7)

11-100: 16.9% (12)

101-500: 12.7% (9)

501-1000: 4.2% (3)

Over 1000: 56.3% (40)

Total responses: 71

20. Number of IT employees/contractors within your organisation.

0-10: 19.7% (14)
11-100: 31.0% (22)
101-200: 7.0% (5)
201-500: 7.0% (5)
Over 500: 35.2% (25)
Total responses: 71

21. Roughly, what percentage of the workforce within your IT department that has engineering qualifications; eg, engineering degree (including software engineering), chartered status, etc?

Over 90%: 19.1% (13)
50%-90%: 13.2% (9)
10%-50%: 42.6% (29)
Less than 10%: 25.0% (17)
Total responses: 68

22. Roughly, what percentage of the workforce within your IT department has a business or management-focused IT qualification?

Over 90%: 5.6% (4)
50%-90%: 8.5% (6)
10%-50%: 39.4% (28)
Less than 10%: 46.5% (33)
Total responses: 71

23. What age group are you in?

18-25: 7.5% (4)
26-35: 58.5% (31)
36-45: 11.3% (6)
46-55: 13.2% (7)
56-65: 7.5% (4)
Over 65: 1.9% (1)
Total responses: 53

The Royal Academy of Engineering

As Britain's national academy for engineering, we bring together the country's most eminent engineers from all disciplines to promote excellence in the science, art and practice of engineering. Our strategic priorities are to enhance the UK's engineering capabilities, to celebrate excellence and inspire the next generation, and to lead debate by guiding informed thinking and influencing public policy.

The Academy's work programmes are driven by three strategic priorities, each of which provides a key contribution to a strong and vibrant engineering sector and to the health and wealth of society.

Enhancing national capabilities

As a priority, we encourage, support and facilitate links between academia and industry. Through targeted national and international programmes, we enhance – and reflect abroad – the UK's performance in the application of science, technology transfer, and the promotion and exploitation of innovation. We support high quality engineering research, encourage an interdisciplinary ethos, facilitate international exchange and provide a means of determining and disseminating best practice. In particular, our activities focus on complex and multidisciplinary areas of rapid development.

Recognising excellence and inspiring the next generation

Excellence breeds excellence. We celebrate engineering excellence and use it to inspire, support and challenge tomorrow's engineering leaders. We focus our initiatives to develop excellence and, through creative and collaborative activity, we demonstrate to the young, and those who influence them, the relevance of engineering to society.

Leading debate

Using the leadership and expertise of our Fellowship, we guide informed thinking, influence public policy making, provide a forum for the mutual exchange of ideas, and pursue effective engagement with society on matters within our competence. The Academy advocates progressive, forward-looking solutions based on impartial advice and quality foundations, and works to enhance appreciation of the positive role of engineering and its contribution to the economic strength of the nation.



The Royal Academy of Engineering promotes excellence in the science, art and practice of engineering.

Registered charity number 293074

The Royal Academy of Engineering
3 Carlton House Terrace, London SW1Y 5DG

Tel: 020 7766 0600

Fax: 020 7930 1549

www.raeng.org.uk