Advances in the Fields Software Engineering Education and Professionalism

J. Barrie Thompson

1. Introduction: The Key Role of Software Engineering

It is widely recognised that Information Society Technologies (IST) are changing the world. For example, in the documentation [FP6] relating to the priority thematic areas for research within the European Union Framework 6 initiative, it is stated that:

"Information society technologies(IST) are transforming the economy and society. Not only are they creating new ways of working and new types of business, but provide solutions to major societal challenges such as healthcare, environment, safety, mobility and have far reaching implications on our everyday life. The IST sector is now of one of the most important of the economy, with an annual turnover of EUR 2000 billion, providing employment for more than 12 million people in Europe."

Also, within the documentation for the Framework 6 initiative, the following needs are emphasised:

- Technologies for trust and security
- Addressing societal challenges
- The development of a trusted knowledge-based society
- New computational models
- The development of new technologies for software and systems that address composability, scalability, reliability and robustness

Simply put, the future will depend on IST, and IST will have a major effect on citizens, business, and organisations throughout the world. Nevertheless, this recognition needs to be taken a stage further-there needs to be a realisation that such a future will depend on those who develop and deliver the systems based on IST, and key amongst these will be Software Engineers. For it will be the Software Engineers who, more than any other group, will have to take responsibility for the quality and effectiveness of these systems. Unfortunately, the problems associated with software projects are legion. Too many of these projects are late, over budget and unpredictable. Often entire projects fail before ever delivering an application or, if completed, fail to deliver the required functionality. Generally the quality of what is delivered falls well below what should be expected from leading edge technologies. Systems that over the last few years have very publicly illustrated the types of problems that can occur have been identified by Tse [Tse, 2000] who highlighted:

- Systems failures regarding freight handling when the Kai Tak airport opened in Hong Kong.
- Denver Airport Automated Baggage system which led to the formal opening of the airport being postponed four times.
- The collapse of the attempt to add hotel and car reservations to American Airlines Sabre System for flight reservations involving a write-off of 125 million dollars.

The software industry, in addressing the problems that are associated with software quality, has attempted to improve matters by concentrating on

two aspects: the product itself (the software) and the processes involved in producing that product. The international standards ISO 9126 and 12119 provide for software product quality (ISO 9126: Information Technology-Software Product Evaluation-Quality Characteristics and Guidelines for their Use, and ISO 12119: Information Technology -Software Packages-Quality Requirements and Testing). While, with regard to process, the relevant international standards are the ISO 9000 series of Standards for Quality Management Systems (in particular ISO 9001: Quality Systems -Model for Quality Assurance in Design, Development, Production, Installation and Servicing and ISO 9000-3: Guidelines for the Application of ISO 9001 to the Development, Supply and Maintenance of Software). Yet despite the promotion of all these standards and approaches the industry is still in a position where low quality software continues to be a major problem. What is very clear from case study literature is that, whether one is concerned with product or process, a third vital ingredient is people. It is thus people rather than product or process that should be regarded as fundamental to any quality regime. Other engineering disciplines, which may be considered analogous to that which supports software, have one prime feature with regard to staffing - that at the appropriate levels they have professional staff who are appropriately educated and formally licensed or accredited as competent within their discipline.

It is clear that Software Engineers will play a vital role in a world more and more dependent on IST. However, it is also clear that Software Engineering needs to be seen as a professional discipline - in line with other traditional branches of engineering. For this to occur there needs to be both educational and professional infrastructures which reflect a true "engineering" ethos. That progress is being made in these areas is the subject of the remainder of this paper. In the following section I outline recent international movements towards Software Engineering education and professionalism. Then in the next four sections I consider in more detail the four most significant of those advances. Finally I present some overall conclusions.

2. International Movements in Software Engineering Education and Professionalism

As stated above, to support Software Engineering as a professional discipline there needs to be both educational and professional infrastructures which reflect a true "engineering" ethos. During the last five years there has been some progress and some clear indications that these areas are receiving higher priority in some quarters. For instance:

- In 1999 and early 2000 a significant number of academic papers promoting areas related to Software Engineering professionalism started to appear in major computing journals. For example, much of the November/December 1999 issue of IEEE-CS Software and the May 2000 issue of IEEE-CS Computer were devoted to this.
- The IEEE Computer Society and the Association for Computer Machinery had in 1998 created the IEEE-CS/ACM Software Engineering Coordinating Committee (SWECC) which was made responsible for co-ordinating, sponsoring and fostering all the various activities regarding Software Engineering within the IEEE-CS and ACM's sphere of operation. This committee then progressed various projects to advance Software Engineering in areas such as standards of practice and ethics, body of knowledge, curriculum guidelines, and exam guidelines. However, in summer of 2000 this formal co-operation came to an end.
- The Texas Board of Professional Engineers had in June 1998 enacted rules that recognised Software Engineering as a distinct engineer-

ing discipline plus legislation that enabled Professional Engineering licenses to be issued to software engineers in Texas. However, subsequent progress has been slow. The number of licences issued has been low and each has depended on examination waivers. Also, such a system of licensing has not yet been adopted by any other USA state.

- A joint task force on Software Engineering Ethics and Professional Practice (SEEPP), established by SWECC under the chairmanship of Donald Gotterbarn of East Tennessee State University, has developed a Software Engineering Code of Ethics and Professional Practice. This is available for public comment on the web and exists in two forms: a short version which summarises aspirations at a high level of abstraction and a full version which includes additional clauses. The latter provide examples and details of how the aspirations of the code should change the way persons act as Software Engineering professionals. The code has been accepted by both the IEEE-CS and the ACM and in addition many other national professional bodies for computing have reacted positively to it.
- The Guide to the Software Engineering Body of Knowledge (SWEBOK) project, which was also initiated by SWECC, has aimed at achieving a consensus view by the Software Engineering community on a core body of knowledge (BoK) for professionals within the Software Engineering discipline. The project is being run from the University of Quebec in Montreal and it is taking a three-phased approach similar to that adopted for the development of the Ada programming language. The second phase of the project, involving more than 200 reviewers resulted in the Trial Version of the Guide (version 0.9) which has been promoted for public use since early 2001. The results of this use are currently being feed into the final phase of the project.
- In autumn 1998, the Association for Computing Machinery (ACM) and the Institute for Electrical and Electronic Engineers Computer Society (IEEE-CS) established a joint task force to undertake a project devoted to producing a new version of their curriculum guidelines for undergraduate programs in computing. The project was named Computing Curricula 2001 (CC2001) and the task force were directed to:

  "To review the Joint ACM and IEEE/CS Computing Curricula 1991 and develop a revised and enhanced version for the year 2001 that will match the latest developments of computing technologies in the past decade and endure through the next decade."

  The early work within the curriculum project resulted in the task force deciding to divide the new Computing Curriculum 2001 (C2001) report into several volumes, each of which would focus on a particular computing discipline. One of these volumes will specifically address Software Engineering and it is expected after public review and feedback to be completed late in 2003.
- During the 1990s the International Federation for Information Processing (IFIP) were progressing a project on the Harmonisation and Acceptance of International Standards for IT Professionals, as it was believed that there was a need for rationalisation in this area. Also, the Word Trade Organisation was promoting the view that, in an era of international treaties which promoted free trade and the free movement of workers from one country to another, the establishment of standards for the qualifications of professionals. In 1998 a draft standard entitled "Harmonisation of Professional Standards" was produced. This document set out the standards of tertiary education, experience or practice, ethics, and continuing education that a customer might expect from a practitioner offering services to the public. It

was hoped that eventually the document could be developed to become a Standard in the sense of ISO. During the last three years a significant number of international activities have been undertaken to promote and evaluate IFIP' s harmonisation document with regards Software Engineering.

·IEEE-CS have recently introduced a scheme to enable it to offer a Certified Software Development Professional designation to successful applicants. The scheme has clearly been designed to support the society's efforts to establish Software Engineering profession. The professional certification being offered by IEEE-CS has three components: exam based testing demonstrating mastery of a Body of Knowledge, extensive experience base in the performance of the work or profession being certified, and continuing professional education. Details of the scheme is available on the web and also on a CD [CSDP]. The latter also contains reprints of related academic articles plus copies of Software Engineering Code of Ethics and Professional Practice and the SWEBOK Guide (Trial Version).

Internationally the most significant of the above are probably:

1. The project concerned with the Software Engineering Code of Ethics and Professional Practice,
2. Guide to the Software Engineering Body of Knowledge (SWEBOK) project,
3. The Production of the Software Engineering volume as part of the CC2001 effort,
4. IFIP's proposals within its harmonisation document and their relevance to the Software Engineering profession.

Each of these will now be considered in turn.

3. Software Engineering Code of Ethics and Professional Practice

The IEEE-CS/ACM Co-ordinating Committee (SWECC) which was described above has been responsible for the creation of a joint task force on Software Engineering Ethics and Professional Practice(SEEPP). This task force, under the chairmanship of Donald Gotterbarn of East Tennessee State University, has developed the Software Engineering Code of Ethics and Professional Practice. The code is available [SEEPP] in two forms: a short version which summarises aspirations at a high level of abstraction and a full version which includes additional clauses. The latter provide examples and details of how the aspirations of the code should change the way persons behave as Software Engineering professionals.

The code is available for public comment on the web [SEEPP]. Currently the eight principles laid out in the short code are:

1. PUBLIC - Software engineers shall act consistently with the public interest.
2. CLIENT AND EMPLOYER - Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
3. PRODUCT - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.
4. JUDGMENT - Software engineers shall maintain integrity and independence in their professional judgement.
5. MANAGEMENT - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
6. PROFESSION - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
7. COLLEAGUES - Software engineers shall be fair to and supportive of their colleagues.
8. SELF - Software engineers shall participate in lifelong learning regarding the practice of

their profession and shall promote an ethical approach to the practice of the profession.

The code has been accepted by both the IEEE-CS and the ACM, and other national professional bodies for computing have reacted positively to it. There is also currently more recognition, within certain quarters, that ethical issues are becoming more important within computing.. The membership of the task force which produced the code, and which consisted of a three-person executive committee and a general membership of 22 members, had a truly international composition. Many of them are active members in computer ethics organisations within their own countries and are also actively involved with international conferences concerned with computer ethics such as the Ethicomp Series [ETHICOMP]. It could be that such an international composition is one of the reasons that the code has been so well received. Nevertheless there have been some problems. The code has of late received some publicity both at conferences and in International Journals. However, the whole area of professionalism tends to be neglected by major parts of the software industry, which appears to be mesmerised with technological aspects to the detriment of human related areas and issues.

There are also problems on how ethics education should be incorporated into the curricula and how it should be taught. Many believe that it needs to be fully integrated into all subjects and that positive guides to ethical behaviour need to be in place in current software engineering courses starting in the early stages and running through the entire program.

4. The Guide to the Software Engineering Body of Knowledge (SWEBOK) Project

The SWEBOK project, which resulted from the co-operation between the IEEE-CS and ACM on Software Engineering, is aimed at achieving a consensus view by the Software Engineering community on a core body of knowledge (BoK) for the Software Engineering discipline. The project is being run from the University of Quebec in Montreal and it is taking a three-phased approach similar to that adopted for the development of the Ada programming language, consisting of Straw Man, Stone Man, and Iron Man phases. Full details of the project and its development can be found on the project' s web site [SWEBOK]. The results of the Straw Man phase were published in September 1998 and defined the project' s strategy and rationale. The major part of the subsequent Stone Man phase has been the production of Knowledge Area Descriptions (KADs) and associated documentation which was then distributed for peer review across the Software Engineering community world wide. To try and ensure that the KADs were relevant to all audiences within the Software Engineering community and the intended usage by these audiences, the reviewers for the documentation have been selected to represent a spread across a spectrum of viewpoints from Individual Practitioners to Trainers and Educators.

During April 2000 a more or less finalised edition of the Stone Man Version of the Guide (version 0.7) was released on the project' s web site. This was then further refined and renamed as the Trial Version (version 0.9) which would be promoted for public use (the name was amended so that users would not incorrectly assume that the contents have been "set in stone"). In April 2001 the Industrial Advisory Board for the project approved field trials for the Guide for a two-year period. During this time the members of the project team have been very active in promoting the Guide which has been made available both as a hard copy international publication and a free download from the project site. A request for formal reviews on the use and content of the Trail Version of the guide was announced in May 2003. It is understood, at the time of writing this paper in August 2003, that these formal reviews are

currently undergoing analysis. The results of this analysis will then feed into the third stage of the project.

The SWEBOK project represents a very systematic piece of work that has attempted a broad and international approach in its reviewing process. Although the production timescales are rather long it should finally produce an authoritative and accepted BoK for the discipline. Of particular note is that the whole of the reviewing process has been visible and is available on the project' s web site. Although the BoK is not yet in its final form the work accomplished so far be can be used to support the teaching within the discipline. The Guide certainly is a very useful resource for both staff and students at undergraduate and graduate levels. Also, the reviews themselves can be used to demonstrate to students the disparate views that exist across the community. However, these very positive features also have a downside. Clear difficulties with SWEBOK arise because of it being an extremely large and complex project with a rather long timetable. It is also likely that it suffered from insufficient publicity on an international level during the earlier stages, though that does appear to have improved over the last three years where there has been a clear effort to publicise it at relevant international conferences and in journals. In addition, a close inspection of the contents of the trial version of the guide shows what could be regarded as a USA, or at least a North American continental, bias within several parts of it.

5. Software Engineering Volume of CC2001

To progress the work on the Software Engineering volume for undergraduate curricula (known as CC2001 Software Engineering or simply CCSE) there is a joint ACM and IEEE-CS initiative supported by four groups of volunteers:

·A Steering Committee, which has responsibility for the organisation and co-ordination of the development of CCSE (on which the author of this paper represents the IEEE-CS Technical committee on Software Engineering).

·An Advisory Board.

·An Education Knowledge Area Group, which is responsible for defining and documenting a Software Engineering body of knowledge known as the Software Engineering Education Knowledge (SEEK). This is a body of knowledge that is seen as being specific to the development of undergraduate Software Engineering curricula.

·A Pedagogy Focus Group, which is responsible for using the SEEK in the development of undergraduate Software Engineering curricula and the definition of undergraduate SE courses/ programmes. It will also consider and advise on appropriate SE pedagogy.

Also, a CCSE web site [CCSE] has been established to support the project and document progress. The CCSE documents have been posted there as and when the various working groups have produced them.

In the development of the volume, extensive use has been made of external reviewing by both the general Software Engineering community and by international experts in the field. Also, of note are a number of open events that have been held at international conferences to assist in the development of the volume. Notable of these have been:

1) A workshop which was held on 25th February 2002 at the Fifteenth Conference on Software Engineering Education and Training (CSEE&T 2002), in Covington, Kentucky. [Thompson and Edwards, 2002]
2) An International Summit on Software Engineering Education (SSEE) which was held on Tuesday 21st May 2002 and was co-located with the 24th IEEE-CS/ACM International Con-ference on Software Engineering (ICSE 2002), in Orlando, Florida. [Thompson et al, 2003].

3) A second International Summit on Software Engineering Education (SSEE II) which was held on Monday 5th May 2003 and was co-located with the 25th IEEE-CS/ACM Inter-national Conference on Software Engineering (ICSE2003), in Portland Oregon [Thompson and Edwards, 2003a]

The project is naturally large, complex, and has an expected life of 10 years or more. Progress at times has been slow at times. However, a first public draft of the Software Engineering Volume was released for public review this August via the public web site [CCSE]. Since then the steering committee have been actively soliciting reviews and it is hoped that the Volume should be finalised by the end of the year. In the current draft the main chapters cover:

·Guiding principles
·The Software Engineering Discipline
·Overview of Software Engineering Education Knowledge (SEEK)
·Guidelines for Software Engineering Curriculum Design and Delivery
·Courses and Course Sequences

A clear reason why the development of CCSE has been somewhat slow is that across North American Universities there are still relatively few Software Engineering programmes in comparison to the surfeit of Computer Science programs. This is simply not the case in other countries. For example, in the UK in 2002 there were at least 50 institutions offering Software Engineering as a single subject degree [Edwards et al, 2003]. The lack of a wide base of existing programs in North America has tended to make the developers of parts of the volume somewhat cautious in their proposals and there has a clear wish to do much from "first principles" rather than take a more "engineering" approach of "improving what already exists". In the development of the volume there have also been issues regarding what actually constitutes undergraduate education. One major issue that became very clear during the SSEE summit in Orlando summit is that there were two very differing viewpoints with regard to this level of education. One which regarded it as primarily about knowledge and the other which rated understanding and ability more highly.

With an expected lifetime for the volume of some 10 years there will need to be mechanisms in place that will ensure, as far as is possible, that any developed curricula remain reasonably current. Also there will be ongoing issues about addressing different modes of learning and delivery as in many parts of the world within the next 10 years traditional classroom delivery may no longer be the norm.

6. Professional Standards Initiative by International Federation for Information Processing (IFIP)

IFIP is a non-governmental, non-profit umbrella organisation for national societies working in the field of information processing. Technical work, which is the heart of IFIP's activity, is managed by a series of Technical Committees (TCs). Each of these Technical Committees in turn is responsible for a number of Working Groups (WGs). For example, WG 3.4 is concerned with professional and vocational education and training in the field of information technology. Full details of IFIP's structure and operation can be found on the organisation's web site [IFIP]. In 1998 a working party within TC3 produced a draft document concerned with the "Harmonisation of Professional Standards" (a copy of which is include as an Appendix to this paper and it can also be found with the record of meetings on IFIP' s site). The draft standard was presented in August 1999 at the overall TC3 committee meeting in Irvine, USA and at the TC3 WG3.4 seminar held in Baltimore, USA. In the draft, introductory sections explain the overall purpose of the work, why professional standards are needed, to whom the standard will apply and clarifications concerning the terminology

used. The main part of the standard then addresses the following areas, all of which are obviously relevant to Software Engineering:

·Ethics of professional practice,
·Established body of knowledge,
·Education and training,
·Professional experience,
·Best practice and proven methodologies and
·Maintenance of competence.

The IFIP harmonisation document does represent a very high level view of what is needed, nevertheless, it is very sensitive to the many complex issues that exist in the area of professionalism. It thus can provide a useful framework for further work regarding Software Engineering professionalism also it has positive attributes in that it was developed with international use as a major goal. Over the last three years conference presentations, panel sessions, participative workshops and summit meetings have been used to promote the harmonisation document within the Software Engineering arena. Most notable have been:

·A workshop which was held at the 2001 Conference on Software Engineering Education and Training (CSEE&T) in Charlotte, North Carolina in February 2001. [Thompson and Edwards, 2001].
·A full day workshop which was held during the 2001 International Conference on Software Engineering (ICSE) in Toronto in May 2001. [Thompson, 2001].
·A full day summit which was co-located with the 2002 International Conference on Software Engineering (ICSE) in Orlando in May 2002.. [Thompson and Edwards, 2003b].

Major conclusions from these activities have been:

·That there is a need for structures to support Software Engineering professionalism and that there is a need for harmonisation.
·That the IFIP document draft is a positive step and that it can be used as a framework or meta model for Software Engineering professionalism. However, more work needs to be done on it at a detailed level.
·That the Software Engineering Code of Ethics and Professional Practice and the Guide to the Software Engineering Body of Knowledge (SWEBOK) can be seen to satisfy the first two area highlighted in the IFIP document.
·That public safety is a fundamental driver towards professionalism. The view was expressed that one day something really major will go wrong and it will be the subsequent legislation produced by countries/states/provinces that will provide the impetus for professionalism.

7. Conclusions

Much of the work that has been done under the auspices of IEEE-CS and ACM high standard and it would be a disaster if this type of work was not carried on. However, Software Engineering is a discipline that must operate at a global level. Other engineering disciplines such as Mechanical Engineering (which shaped the 19th century) or Electrical Engineering (which shaped the 20th century) to a great extent developed and operated within domains defined by nation states or, at least, continental boundaries. Software Engineering is different to these older disciplines in that it must be viewed in a wider context. Already we have situations where, for example, software can be specified in the USA, developed in India, and then used globally on the Internet. It is thus of paramount importance that the Software Engineering discipline is viewed at a global level rather than at just at the continental or national level.

The IFIP report on the Harmonisation of Professional Standards should encourage international co-operation within all computing disciplines including Software Engineering. Also the very

nature of the IFIP committees ensures that it has, and will be, considered at an international level and hence can not be judged to represent the view of only one sector or country. The harmonisation document essentially defines a framework, which should truly assist advancing Professional Standards if it is used in a sensitive and appropriate manner. If we consider the six areas addressed in the IFIP document and the ACM/IEEE-CS supported projects discussed in this paper we can see that there has been real progress in the fields of Software Engineering education and professionalism:

- ·The ethics of professional practice is supported by the Software Engineering Code of Ethics and Professional Practice.
- ·An established body of knowledge is provided by the Guide to the Software Engineering Body of Knowledge (SWEBOK) and also by the Software Engineering Education Knowledge (SEEK) defined in the CCSE Volume.
- ·Education and training needs are supported by the CCSE Volume though this is primarily directed at the former.

Nevertheless, there is much that still needs to be done with regard to:

- ·Professional experience and training
- ·Best practice and proven methodologies
- ·Maintenance of competence

Therefore it is planned that to address issues in these areas a further summit (or similar event) will be held at the IFIP 2004 World Computer Congress 2004 in Toulouse during August 2004. It is hoped that such an event will enable opinions to be formed on:

- ·What are the key steps in a career in Software Engineering and other branches of computing?
- ·What education and training are really needed?
- ·How should professional behaviour be regulated?
- ·How is competence maintained and certified?

Acknowledgements

Parts of this invited paper were developed from the publications that I presented at Fifth International Conference on The Social and Ethical Impacts of Information and Communication Technologies (ETHICOMP 2001) held in Gdansk in June 2001, the Twenty- Sixth Annual International Computer Software and Applications Conference (compsac 2002) held in Oxford in August 2002, and the 2002 Asia-Pacific Software Engineering Conference (APSEC 2002) held in the Gold Coast in December 2002.

References and Sources of Further Information

CCSE, Computing Curricula Software Engineering - the Software Engineering volume of CC2001, information is available from the project's web site:http://sites.computer.org/ccse/

CSDP, IEEE Computer Society Certified Software Development Professional, Web Site: http://computer.org/certification

Support CD: Developing Software Engineering as a Profession, Published by Institute of Electrical and Electronic Engineers, Inc, ISBN 0-7695-1899-0.

Edwards H. M. Leckenby B. M. and Thompson J. B. (2003), A census of Software Engineering Undergraduate Programmes in the UK, awaiting publication, Further information available from: Software engineering Research Group, University of Sunderland, UK.

ETHICOMP: International Conferences on Ethical Issues of Information Technology, Fourth held in Rotterdam March 1998, Fifth held in Rome in October 1999. Sixth in Gadansk in 2000, Seventh in Lisbon in 2002, Details available from Centre for Social Responsibility, De Montfort University, UK.

FP6, Sixth Framework Programme of the European Community for research, technological development and demonstration activities, The priority thematic areas of research in FP6 are

detailed in Annex 1 of the documentation for Expressions of Interest and are held at:
ftp://ftp.cordis.lu/pub/fp6/eoi-instruments/docs/eoi_annex1.pdf

IFIP International Federation for Information Processing information available from web site:http://www.ifip.or.at

SEEPP Project concerned with the Software Engineering Code of Ethics and Professional Practice. Details of this are available at the following Web Sites:
http://computer.org/tab/seprof/code.htm, and
http://computer.org/tab/sweec/SWCEPP

SWEBOK, Software Engineering Body of Knowledge Project , Web Site is at:
http://www.swebok.org

Thompson J. B. (2001) Report on the ICSE Workshop to Consider Global Aspects of Software Engineering Professionalism, which was held on 14th May 2001 in Toronto, ACM SIGSOFT Software Engineering Notes, November, 2001.

Thompson J. B. and Edwards H. M. (2001), Report on the Workshop on Achieving a World-wide Software Engineering Profession" which was held at Fourteenth Conference on Software Engineering Education and Training in Charlotte, North Carolina in February 2001, Journal of Education and Information Technologies, 6:4, December 2001, pp.267-293.

Thompson J. B. and Edwards H. M. (2002), Preliminary Report on the CSEET 2002 Workshop "Developing the Software Engineering Volume of Computing Curriculum 2001", Which was held on 25th February 2002 at the Fifteenth Conference on Software Engineering Education and Training (CSEE&T 2002), in Covington, Kentucky, Forum for Advancing Software Engineering Education (FASE), Volume 12 Number 03 (Issue 146), March 15, 2002.

Thompson J. B. and Edwards H. M. (2003), Report on the 2nd International Summit on Software Engineering Education, Co-located with ICSE2003, which was held on Monday 5th May 2003 and was co-located with the 25th IEEE-CS/ACM Inter-national Conference on Software Engineering in Portland Oregon, accepted for publication in ACM Software Engineering Newsletter.

Thompson J. B. and Edwards H. M. (2003), Post-Summit Proceedings International Summit on Software Engineering Professionalism (SSEP), which was held on Monday May 20th 2002 and was co-located with the 24th IEEE-CS/ACM International Conference on Software Engineering (ICSE2002), in Orlando, Florida), available from Learning development Services, University of Sunderland, ISBN: 1-8737757-34-4.

Thompson J. B. Edwards H. M. and Lethbridge T.C (2003), Post-Summit Proceedings International Summit on Software Engineering Education (SSEE), which was held on Tuesday 21st May 2002 and co-located with the 24th IEEE-CS/ACM International Conference on Software Engineering (ICSE2002),in Orlando, Florida, available from Learning Development Services, University of Sunderland, ISBN: 1-8737757-34-4.

Tse, T.H., (2000), Exemplars from presentation in support of the position paper: Towards Harmonized Professional Standards for Software Engineers: Constraints, Conflicts and Con-cessions, Within the panel session entitled: Is the Harmonisation of Professional standards for Software Engineers Feasible? Or Even Welcome?, 24th Annual International Computer Software and Applications Conference (compsac2000), October 25-27, 2000, proceedings published by IEEE-CS, Los Alamitos, ISBN 0-7695-0792-1, pp.345-347.

APPENDIX

INTERNATIONAL FEDERATION FOR IN-FORMATION PROCESSING
Harmonization of Professional Standards
Draft: October 1998

Summary

This document sets out an international standard for professional practice in information technology.

Practitioners who meet the standards will:

- publicly ascribe a code of ethics published within the standard.
- be aware of and have access to a well-documented current body of knowledge relevant to the domain of practice.
- have a mastery of the body of knowledge at the baccalaureate level.
- have a minimum of the equivalent of two years supervised experience before the practitioner operates unsupervised.
- be familiar with current best practice and relevant proven methodologies.
- be able to provide evidence of their maintenance of competence.

Purpose

The purpose of this work is to clearly set out an international standard for professional practice in information technology.

The components of the standards are:

- Ethics of professional practice,
- Established body of knowledge,
- Education and training,
- Professional experience,
- Best practice and proven methodologies and
- Maintenance of competence.

A customer has a right to expect that a practitioner offering information technology services to the public meets these standards.

This document will be offered as a draft standard to the International Standards Organization in anticipation that it will in turn conduct its process of obtaining consensus from its member bodies and hence the standard would be adopted by the standards bodies within each country.

It is expected that the IFIP member societies would prepare any local or regional adaptation of the standard. The administration process, which may include promotion, assessment and certification as well as the distribution of materials, may also be carried out by the IFIP member society.

The standard could also be incorporated in the requirements for a level of qualification of individual members in the member society.

Although the initial country or regional implementations may have differences, the intent is to move towards a common implementation.

Why Have Professional Standards?

The traditional professions such as accounting, medicine and engineering have long had standards which enable a qualification gained in one country to be recognised in another. The World Trade Organisation in conjunction with the International Standards Organisation has now taken an active role to create such standards under the General Agreement on Trade in Services (GATS).

The benefits of internationally recognised standards are that:

- the public is assured that safety or economically critical work is performed by competent individuals regardless of where in the world those persons gained their qualifications and experience.
- a client is assured that a person who meets such international standards is competent to carry out tasks in documented specific areas regardless of where the work is done or the output of the work is used (subject to recognition of issues of culture and locale).
- professionals are assured that their qualifications if recognised in one country will be accepted in other countries without re-examination (except possibly for being up-to-date).
- Under GATS, trade in products developed by practitioners who meet this standard cannot be restricted on the grounds that the developers were not competent or used inadequate professional practices.

Such standards will contribute to the attainment of a reputation for competence by the profession.

The standards will facilitate the obtaining of work by individual practitioners in the international arena.

To Whom does the Standard Apply?

This standard is primarily focused on practitioners involved in the development of software-based systems and related services. The standards are not necessarily intended to apply to other members of IFIP member societies such as:

- academics, who in general will be much more qualified but possibly in a narrow discipline and whose research may be at a more abstract level than practice.
- school teachers, who in general will be qualified to teach rather than to develop IT systems.
- users, who have input into the designs of computer systems but who do not construct them.
- electronic engineers, who design computers but who would normally be qualified as engineers.

It is recognized that these classifications may be blurred.

Harmonization of Professional Standards

The following clarifications are offered in this context.

Harmonization means that the standards of different countries would be brought together to be substantially the same. Any extremes from the commonality of these standards would gradually be pruned away until each country has the same standard by mutual consent.

Professional identifies the peculiar responsibility of a person with high levels of knowledge and related practical skills in a given discipline with respect to members of the public who do not have that knowledge or skill-set. It is particularly relevant to the information technology profession because it has significant impact on society at large. The power of the knowledge must be balanced by a sense of responsibility towards others. This definition is focused on practitioners, persons who actually develop, maintain and operate software systems for commercial or governmental purposes.

Standards are clear statements that reflect the minimum qualifications for mastery and knowledge of processes, skills and practice that a professional should have before undertaking work which may put an employer or client at risk, either physical or financial.

The field of Information Processing has many domains ranging from data management to embedded software systems. Any one individual cannot be expected to be expert in more than one or a few such domains. This needs to be recognized particularly in the body of knowledge required to be known by one person.

The changes within the many domains together with the dynamic development of new domains in information technology means that the standards themselves must be continuously developed and individuals must anticipate life-long learning.

The Standard for Professional Practice in Information Technology

Ethics of Professional Practice

A code of ethics acknowledges the professional responsibilities of practitioners to society at large, members of the public, employers, contracting parties and fellow practitioners.

Codes of ethics have been published by many member societies and IFIP itself.

Every implementation of the standard must include a code of ethics.

Such a Code of Ethics must be compatible with the culture of the society in which the practitioner normally works.

Practitioners must operate in a manner compatible with the culture of the locale in which they are currently working and in which the product may be used.

Practitioners must publicly ascribe to the code of ethics published within the standard.

Established Body of Knowledge

Several IFIP member societies have published bodies of knowledge, some of which have gained wide acceptance. Such recognised bodies of knowledge are divided into many domains determined by the various services carried out by practitioners. The body of knowledge on which any implementation is based should include at least the common components of these but also ensure that each domain is complete in itself for the domains adopted locally.

Mastery of such a body of knowledge forms the basis of preparation for practice. A practitioner must demonstrate mastery of at least one such domain as well as all core components identified in the body of knowledge.

Practitioners must be aware of and have access to a well-documented current body of knowledge relevant to the domain of practice.

Education and Training

Most practitioners will enter the workforce with prior education and training which will commonly be a baccalaureate degree assessing the mastery of the body of knowledge.

Institutions offering such education and training should be prepared to openly compare themselves to internationally well-known and recognised peer institutions offering similar programmes.

It is recognised that this level of mastery may be achieved by various combinations of education and experience. Nevertheless a practitioner must be able to provide evidence of such mastery to practitioners who have met this standard.

The minimum level of mastery of the body of knowledge must be at the baccalaureate level.

Professional Experience

Experience builds on knowledge in many essential ways. Such as:

- It develops and improves practical skills and competencies.
- It provides understanding of task definition in the users' terms.
- It helps develop interpersonal skills that facilitate the communication and human interaction between all participants.
- As many approaches to problem solution are not readily scaleable experience over a wide variety of problem types and sizes is desirable before working in an unsupervised environment. Experience is generally required in assessing task complexity.
- Task management, overall project management and quality management generally require experience.

Other professions have clear requirements for experience before allowing their members to practice without supervision.

In addition to a demonstrated mastery of the body of knowledge a minimum of the equivalent of two years supervised experience is recommended before the practitioner operates unsupervised.

Best Practice and Proven Methodologies

Experienced practitioners have identified and documented many practices and methodologies the use of which generally leads to successful project outcomes. Where such best practice and proven methodologies are available the practitioner should use them unless a particular task has exceptional attributes.

Member societies drawing on all available international sources should encourage the documentation and promulgation of best practice and proven methodologies.

Practitioners should be familiar with current best practice and relevant proven methodologies.

Maintenance of Competence

To maintain demonstrated competence practitioners must be familiar with new developments in their domains of practice.

Such developments may be reflected in the body of knowledge, best practice and proven methodologies as well as in specific skills.

Familiarity with new developments may be obtained through formal education or peer interaction.

There may be assessment of current competence by formal examination, peer assessment or employer or client acknowledgement of successful work.

A practitioner should participate for at least the equivalent of 10 days per year in activities that contribute to maintaining competence. It is recognised that in different locations the opportunities for such ongoing development may vary.

The standard in each country or region must state how this requirement will be met and the role of the IFIP member society in monitoring this function.

Practitioners must be able to provide evidence of their maintenance of competence.

Drafted by Ian Mitchell, FNZCS, Peter Juliff, FACS and Joe Turner, FACM.

J. Barrie Thompson

Professor in Applied Software Engineering at the University of Sunderland, United Kingdom,

Co-leads with Prof. Helen M. Edwards the University's Software Engineering Research Group.

Vice-chair of Working Group 3.4 (Professional Education and Training) within the International Federation for Information Processing.

Representative for IEEE-CS Technical Committee on Software Engineering on the Steering Committee for Computing Curricula Software Engineering.

Major research interests are: educational, professional and ethical aspects associated with area of Software Engineering, the use of methods and Software Engineering principles in the development of commercial/administrative computer based information systems, the development of software tools and quality assurance approaches to support commercial software development

Publications and presentations total more than 160 of which more than 80 have consisted of published textbooks, referred Journal and Conference papers, chapters in published books, and the texts of Invited International Seminars.