Submission to ICSE'2000 workshop entitled "Beg, Borrow or Steal: Using Multidisciplinary Approaches in Empirical Software Engineering Research", June 2000, Limerick, Ireland

Improving Information System Success using Cooperative and Multidisciplinary Development Techniques.

Mark Maritz*, **Rachel Harrison****
*Visiting Research Fellow, M.Maritz@reading.ac.uk, **Rachel.Harrison@reading.ac.uk
School of Computer Science, Cybernetics and Engineering, University of Reading,
Reading, United Kingdom, +44 (0)118 931 8617

ABSTRACT
This position paper seeks to describe the problems that might benefit from a multidisciplinary approach. The paper specifically focuses on the challenges of communication between the academic and industry community. A broad range of skills is required from software engineering team members as they seek to produce relevant software solutions that are timely and of good quality. A multidisciplinary approach is seen to be vital to the research process in the above environment as well as to the software development process itself.

Keywords: Multidisciplinary, Software Engineering Success, Organisational Effectiveness, User Satisfaction, Effectiveness, Efficiency, IS Success Model.

INTRODUCTION
Software engineers face incredible challenges in the new millennium. The Internet and the increasingly demanding Electronic Business revolution are requiring engineers to produce software products far more quickly and effectively. Engineers are also required to be more flexible in their approach while having the ability to communicate amongst themselves as well as with the users in an organisation. Various factors are now seen to play a role in the success of software projects and the ultimate aim of a development project is now not just the delivery of a software solution, but a solution that will ultimately have an organisational impact.

Universities have a crucial role to play in the training and development of graduates as well as the delivery of relevant and meaningful research that is useful to business and industry. Graduates need to have skills that are relevant to the software development industry as a whole.

INFORMATION SYSTEMS, COMPUTER SCIENCE AND SOFTWARE ENGINEERING
The Information Systems (IS) discipline, views computing from a business perspective, where amongst others, management, communication, and application issues are the focus. The traditional waterfall model of the SE process [4] has the following stages; requirements analysis and definition, system and software design, implementation and unit testing, integration and system testing, and operation and maintenance.

"Software engineering (SE) is the science and *art* of specifying, designing, implanting and evolving – with economy, timeliness and elegance – programs, documentation and operating procedures whereby computers can be made useful to man." Computer Science (CS) may be seen as "the systematic study of algorithmic processes that describe and transform information: their theory, analysis, design, efficiency and application"[3].

The reference to SE being an *art* is almost critical to the SE process described above and focuses the attention to the fact that the process does not and cannot be totally scientific in nature. It involves many skills and techniques that may be very unscientific in nature. Skill examples may include; communication, management, delegation and negotiation skills.

These skills are often not transferred to, and developed in, SE graduates during their studies and may require experience to acquire and master, for example the understanding of group dynamics. Most SE students do have the opportunity to work in groups during their studies but these groups are often solely made up of other SE students. Ideally this should not be the case and multidisciplinary groups need to be created that allow students from different disciplines to work together on projects.

A group may then be made up of commerce, social science, SE and CS students, which would allow for a far broader skills base and a more realistic learning environment. There are however many problems associated with implementing such a multidisciplinary project as Ian Sommerville observes. He notes that that 50% of the process involves interaction between the programmer and other team members [4]. This is an important statistic, as it points to the need for communication skills on the part of the developer.

A comment made by a professional in a private or public company, in a study by Hatling, et al [2] clearly illustrates the attitude of some professionals to the "softer" issues associated with the systems development process: He believes "the information system you have to have. While the other stuff [Knowledge about organisations] is not necessary." Even more of a concern is the fact that the professional recognises the soft skills but deems them to be unnecessary.

A second respondent, in the above study, believes that the organisational issues should be resolved on a management level and not be the concern of the development team [2]. This comment again highlights the very low importance that is given to the "softer" issues associated with a SE project by many SE professionals.

There can be little argument with the fact that software engineering involves large amounts of individual talent and ability on the part of the developer. A concern is that many software engineers do not have the time or the inclination to consult with the business users of the software and visa versa.

THE SE COMMUNITY, INDUSTRY AND SOCIETY

In terms of ethics, the SE community has various obligations to society, to their employers, their clients, and their colleagues and professional organizations. I believe one of these should be relevance; relevance to society, industry and to the future. To do this the SE community needs to find solutions and develop strategies to allow them to achieve various goals.

The SE community needs to:

1. Contribute to, and accept input and direction from other disciplines,
2. Contribute to industry by making available good quality and relevant research output, while also forging closer relationships with industry during the software development process,
3. Consider the social and economic impact of their profession. Issues such as information quality, system quality, efficiency and timeliness are issues that should be paramount.

SE AND ITS RELATIONSHIP TO INDUSTRY

One of the key problems currently experienced within the SE arena is that of the relationship or lack thereof of the SE/CS academic community to industry. The question that arises is that of the relevance of the output of academe to industrial situations and problems. Does industry actually read academic publications and take note of and use their recommendations? In many instances there seems to be little of interest to a busy manager or developer employed in industry and when there is something relevant this information is often hidden in large amounts of technical data and discussion.

How industry and academe tackle this problem is largely dependant on their ability to build a successful and dynamic SE and industrial community. This community needs to have a well developed and maintainable communication channel built on trust and mutual understanding. The community would almost definitely benefit to both industry and the academe could prove to be of great value as the technological and business environment changes so rapidly. This SE and industrial community can in many ways be developed using the guidelines and ideas described in the IS Success Model as put for ward by Delone, et al.

IS SUCCESS MODEL

In 1992 DeLone, et al[1], developed a model of Information System Development (ISD) success (Figure 1) that identifies specific 'streams' that that are important in the ISD process.

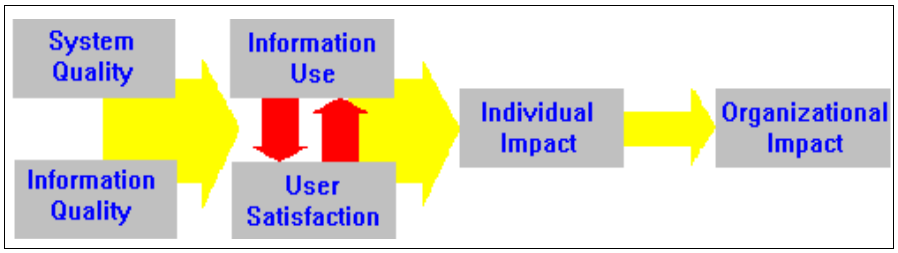

Figure 1: IS Success Model [1]

The various streams can be described below and used to draw attention to the many varying areas that require attention before an ISD project is recognised as a success.

System Quality

The system quality is a measure of the actual system and describes the various issues that relate to its quality, stability and efficiency.

Information Quality

This stream deals with the actual output of the system. Issues that are important in this stream include quality, appropriateness, usefulness and the relevance of the information output.

Information Use

No information system is performing when the intended group of users within an organisation or business does not use the information that is produced. It is therefore critical to the success of the system that there is agreement as to the information requirements of the users. The system will fail if this objective is not met even though the system quality may be of a very high standard.

User Satisfaction
End user satisfaction is only ensured when the information supplied by the system is in the correct format and structure. Information needs to be readable, and concise, for example. The challenge is therefore to supply information that results in a positive user response.

Individual Impact
The information use and user satisfaction has an impact on the individuals within the organisation and it is critical that this impact is seen to be improving the effectiveness and the efficiency of individuals. Individuals that are not necessarily direct users of the system need to be positively effected. Any negative effect within the organisation may well ultimately effect the successful implementation of the new system within an organisation or environment.

Organisational Impact
The ultimate goal of any system is however the organisational impact of the system. Within a business context the organisation should benefit from the introduction of the new system and along with the individual benefits there should be some improvement in the efficiency and effectiveness of the organisation. Savings may take the form of cost reductions, improved time management and use, resource savings and an improved working environment for employees.

Given the above factors, it is therefore vital to the success if an ISD project that a multidisciplinary approach be followed. The project team needs to comprise people from various disciplines and with various skills.

AREAS IDENTIFIED THAT REQUIRE FURTHUR RESEARCH

Further research could be undertaken in the following broad areas:

- Much work is currently being done using Activity Theory in the HCI and IS fields. There is therefore also an opportunity to use the multidisciplinary approach associated with this theory to describe and improve current practice in the SE discipline.
- The relevance of the SE research, to industry and the 'real' world, compared to the productivity of the researcher. Is there a proliferation of research that is not necessarily meeting the needs of industry? This is not a condemnation of innovative and 'blue sky' research!
- What is the role of the software engineer in the Business Needs Analysis process within an organisation?
- Is there a place for the integration or inclusion of the day-to-day users of the system into the IS development team. Research methods and user requirements analysis techniques that could be used include; action research, observation, focus groups, and Joint Application Development (JAD).
- Does the existence of business users, or industry specialists in a development team improve the quality, and the timeliness, of the system ultimately delivered?
- Does an "in-house" development team outperform an external SE development team in terms of IS success?
- What tools and techniques can be used, or need to be developed, to identify real world problems and create software solutions, before the problems disappear or evolve?
- How do the management and/or leadership style of the ISD project leader affect the ultimate success of the project?

CONCLUSION

Various challenges face university research and teaching staff. Solutions are required for a multitude of software engineering and other technology challenges. University SE, CS and IS departments as well as academics need to see themselves and only part of a process that requires communication and collaboration with industry and in many instances government. Academe must endeavour to work more closely with industry by forming alliances so that skills are more efficiently used at all levels of the software development process and so that academic output is more focused on current challenges.

REFERENCES

1. DeLone, W.H. and McLean, E.R. "Information System Success: The Quest for the Dependant Variable", *Information Systems Research*, Vol. 3, No. 1, March 1992.
2. Hatling, M., Monteiro, E., and Sørensen, K.H. "Coping with Interdisciplinarity in Systems Design" [Online] Available: http://www.ifi.uio.no/iris20/proceedings/15.htm
3. McDermid J.A. and Bennett, K.H. "Software engineering research: A critical appraisal" in *IEE Proceedings*, Vol. 146, No. 4, August 1999, pp 179-186.
4. Sommerville, I. "Software Engineering", Addison-Wesley, 1992, pp7, 31.

ENDNOTE

Mark Maritz is currently on sabbatical at the University of Reading from Rhodes University, East London, South Africa. He will return to his lecturing position in the Department of Information Systems during July 2000. His contact email address will then be m.maritz@ru.ac.za.