

# Ethical Risk Management

**Prof. Simon Rogerson**

**Prof. Don Gotterbarn**

**Originally published as ETHicol in the IMIS Journal Volume 15  
No 3 (June 2005)**

On 10 May 2005 Helen Beckett reported in Computer Weekly that, "Most IT directors do not know the full business risks associated with delivering IT services, according to a survey of 178 UK companies. The research found high levels of ignorance and low levels of confidence among IT managers in a critical part of their job. Some 60% said they were unsure how accidents or failures resulting from changes such as IT upgrades or office moves would affect the business." The findings of this survey are indicative of a widespread problem in every country where IT is used by organisations.

Here are five real examples of risky systems.

A databases system used to collect nationwide property data to identify trends in house prices and sales. There was a flaw in underestimating the significance of data integrity for profiling housing markets.

A Maori genealogy database system was used in asset/wealth redistribution. There was a lack of recognition that this was socially sensitive data related to living people.

An administration system was used in the healthcare management of senior citizens. The design was excessively technical and was unfit for purpose at best and totally unusable at worst.

Electronic voting was proposed for use in national political elections. The proposal had over 100 problems associated with acceptability, usability and security.

A data warehouse system was developed by a major software company. The system contained procedural errors which would result in major financial losses for the company.

Such risks are all ignored by traditional quantitative risk assessment used for IT projects and the purely quantitative approach to risk fails to alert developers to significant negative project impacts. Such risks cannot be turned into the quantitative measures related to time or money so often expected by organisational management in their desire to understand the risks associated with the latest systems development project. Quite simply systems development is a risky business that is out of control. Part of the problem is a narrow approach to risk assessment.

Research and practical project experience has shown that limiting risk assessment to purely quantitative risk analysis does not address a broad range of project risks including social, professional and ethical negative project impacts. A new type of qualitative risk analysis, based on engineering environmental impact statements has been developed to complement purely quantitative approaches to software risk. The Software Development Impact Statement process (SoDIS) extends the concept of software risk in three ways: it moves beyond the limited approach of schedule, budget, and function, it adds qualitative elements, and it recognises project stakeholders beyond those considered in typical risk analysis.

As the types of risks increase the range of stakeholders that need to be considered also expands. Using this expanded risk analysis reduced or eliminated the impacts of many previously undetected risks of software development. The successes of the SoDIS process, in the examples cited above, provide strong evidence that a significant side-effect of narrowing project objectives and ignoring their social ethical and professional impacts is a root cause of the failure to understand risk properly (as reported by Helen Beckett) resulting in IT project failures.

The primary focus of developers is on the project development vision defined in terms of budget and schedule overruns and not on satisfying the customer by meeting technical requirements. The common intra-project risks are evaluated and managed using quantifiable values. The result is that the IT industry is still littered with project failures which are due in part to an institutionalised narrowing of the scope of a project's objectives and of the vision to development objectives. This explains why Beckett reports that, "Following change, 72% of IT managers were not completely confident that their business had a good understanding of the new technology assets installed."

Frequently the primary goals of project development - satisfy the customer, deliver the project within budget and deliver the project on schedule - are reversed. The focus of the risk analysis and mitigation narrows to those many issues which impact these goals negatively and risks that would derail the project's development. Often systems are evaluated in terms of the number of faults per 1000 lines of code rather than the side-effects these faults may have on system users or those affected by the system. These may be interesting numbers but they are totally misleading in their specificity.

Research done in the UK, New Zealand and the USA indicates that these inherent problems of existing risk analysis can be addressed in several ways including:

- expanding the list of generic risks
- maintaining focus on the broader project goals
- extending the list of considered project stakeholders

For a development project to succeed, risk resolution should consider:

- the delivered project type, consisting of sector and application
  - the sector within which the software will be used
  - the type of application that is to be addressed
  - the surrounding circumstances of the application
  
- project impacts on all stakeholders
  - direct stakeholders - developers, customers, and others with a business interest indicating intra-project risks
  - indirect stakeholders - users, others whose life circumstances may be impacted by the product, and the social and natural environment indicating extra-project risks
  
- the different stakeholder expectations regarding how to judge a project as a success or a failure

Responsible risk analysis requires categorisation and description of the delivered project, and the associated direct and indirect stakeholders. The extension of risk analysis to a broader range of stakeholders is a necessary but not sufficient condition of adequate risk analysis. Limiting analysis to purely quantifiable risks would still miss many potentially negative impacts which cannot be easily quantified. There is a need to also focus on the broader impacts of the software.

The SoDIS process provides a mechanism for expanding the stakeholders considered and places particular focus on qualitative risks through the use of structured questions. The SoDIS process has been tested on software development in organisations with different location, size, function, scope, development methodology, and technology level; from small projects in consulting companies to projects as large as the UK's scheme for electronic voting. In one case, risks were identified which could have saved the company \$250,000 USD.

The SoDIS process consists of four stages:

1. the identification of the project type together with immediate and extended stakeholders in a project

2. the identification of the tasks in a particular phase of a software development project
3. the association of every task with every stakeholder using structured questions to determine the possibility of specific project risks generated by that particular association
4. completing the analysis by stating the concern and the severity of the risk to the project and the stakeholder, and recording a possible risk mitigation or risk avoidance strategy

The SoDIS audit process identifies significant ways in which the completion of individual tasks of an IT project may negatively affect any of the project stakeholders. It identifies additional project tasks and changes in existing project tasks that may be needed to prevent any anticipated problems. The resulting document, which complements a quantitative analysis, is a software development impact statement which presents all types of potential qualitative risks for all tasks and project stakeholders.

The SoDIS is the missing element in current risk analysis which primarily focuses on some of the quantitative intra-project relationships between selected tasks and selected stakeholders that constitute an IT project. A responsible professional risk analysis examines both the quantitative and qualitative associations between tasks and project's internal and extended stakeholders. To leave out either the quantitative or the qualitative analysis results in unidentified and, worse still, unaddressed risks and project failures.

In efforts to continue improving the SoDIS risk assessment process and educate practitioners, the Software Development Research Foundation ([www.sdresearch.org](http://www.sdresearch.org)) (<http://www.sdresearch.org>) regularly works with companies to audit software projects. If you are interested in having your project audited please contact Simon Rogerson on [srog@dmu.ac.uk](mailto:srog@dmu.ac.uk).

Please send your views on ethical and social responsibility issues and cases of ethical dilemmas to:

Professor Simon Rogerson  
Director  
Centre for Computing and Social Responsibility  
Faculty of Computing Sciences and Engineering  
De Montfort University  
The Gateway  
Leicester  
LE1 9BH  
Tel:(+44) 116 257 7475

Fax:(+44) 116 207 8159

Email:<srog@dmu.ac.uk>

Home Page: ( <http://www.ccsr.cse.dmu.ac.uk>)