IT Risk Management
Information Systems Failures

References

References


• Nulden, M. 1996. "Escalation in IT projects: can we afford to quit or do we have to continue?", Working Paper, Sweden, Goteborg University.


Risk Management

• “Waltzing with Bears” reference

• applied management common sense

– risk management principles presumably covered elsewhere in massive detail

IT Risk Management Process

1. risk census: identify all risks (!!!)
2. include all core risks
3. analyze each risk
4. make some key assumptions re “showstoppers”
5. initial estimate on a no-risk basis
6. construct a risk profile:
   1. 0% confidence delivery date
   2. 20% confidence delivery date
   3. etc.
7. monitor all risks and execute contingency plans if needed
8. maintain a risk discovery process throughout the project
Commitment Date

- targeting the 0% confidence date (the zero risk allowance date) is not sensible for obvious reasons
- the organization will not (should not) allow the 100% confidence date
  - pragmatic is the 75% confidence date
  - this is because levels of uncertainty about IT projects are so high
- for a project of 1 year, the “window of uncertainty” (ie. from 0% to 100% confidence) can be up to two more years (200%)

Include Core Risks

the set of risks identified should include the “core” IT risks as follows:

1. flawed schedule
   (ie. the initial schedule/estimate has a fundamental errors in it)
2. scope creep
3. employee turnover (including resignations and illness)
4. specification breakdown
5. poor productivity

IT Risk Management

- theoretically simple
- in practice managers can find it difficult to determine what constitutes a serious risk and what doesn’t
Why Study System Failures?

- public and private “faces” tend to be different in all aspects of social life
  - theory (and post-hoc rationalization) is the public face, practice the private face
  - lifting the lid on practice can reveal something quite different to what is predicted or implied by theory
- reviews of failures and disasters are usually the only time that the lid is completely lifted.

Systems Failure

- terminology is important
- we are using the term “systems failure” in the broadest sense to refer to
  - situations where a system of some sort (not just an information system) fails to deliver the expected result
    - e.g. a project team acts as a “system” designed to deliver an IT application

The Circumstances of Failure

- what we are really interested in is how things get to the level where “failure” is an appropriate term to use
  - the term is of course a pretty loaded one and not everybody will agree on whether something has been a failure or not
Information Systems Failures: short-listed for fame

- London Ambulance Service
- the “Taurus” System
- Therac-25
- the City Link Billing System (Victoria)
- Intergraph (Victoria)
- “Socrate” - French Railway ticketing system
- INCIS - Integrated National Crime Information System (NZ)
- RMIT Peoplesoft (Victoria)
- SPRINTEL (Italy)
- Teleco BPR
- CONFIG

From Failure to Disaster

- It is rare for an IT failure (though “failures” are common) to turn into a disaster
- but some do (“Normal Accident Theory” - Perrow)
  - experiencing a certain number of catastrophic accidents is “normal” as systems become more complex and interrelated
  - every change has some potential to cause unforeseen problems
  - the causes of accidents are often obvious in hindsight but…
- hard to foresee because it usually “involves failures in several parts of a system which are linked in complex and subtle ways”

Susceptibility to Problems

- the following discussion is relevant to various types of IT initiative that are particularly susceptible to problems
  - software development
  - enterprise systems purchase and implementation
  - software customization
  - e-business support provisioning (eg - b2b procurement platforms)
  - strategic planning
Runaway Projects

• this is a very characteristic type of IS failure - runaway projects are quite frequent in practice

• runaway projects are those which continue past the point at which completion was expected, and then keep going...and going...and going
  – nobody seems to have the nerve or the will to stop them
  – they are no longer economically justified

Taurus - the London Stock Exchange

• this was a systems development which ran from 1986-1994 and failed to deliver a working system

• estimated loss of 50M pounds

• the failure was so bad that various conspiracy theories have been developed
  – sabotage by the “gentlemen’s club” at the Bank of England?

• designed to automate trading processes

• there were no clear lines of control for dealing with requirements
  – London Stock Exchange
  – Stockbrokers
  – Bank of England
  – various software developers (in the USA)

• the final design was based on a software package requiring ~ 40% modification (what is the customisation “limit”?)

• the project was managed (not “steered”) during its later phases by a committee
The “CON FIG” Project

- the paper defines escalation as “the continued commitment of resources in the face of negative information”
- the project commenced in 1981 and was finally terminated in 1994

The “Therac-25” Disaster

- over time, a number of patients were administered lethal doses of radiation by the “Therac-25” machine
- there was a reliance on software for the safe operation of the machine
- a highly skilled operator could in some circumstances initiate the next action before the system had time to process the feedback
- the IS was poorly documented - had been written by one programmer no longer with the company

The “Therac-25” Disaster

- testing had not revealed any problems and the company did not believe that the patients’ injuries were due to a system malfunction
- there were no standards for the reporting and management of problems
- no clear lines of control for the resolution of user difficulties
  - it took years for compensation to be paid…
IS Problems and Failures

• IS failures always occur in a broader systems context
  – ASMA statistics indicate the failure rate exceeds 50% for systems taking more than 18 months to develop

• runaway projects (ie - projects where a development schedule gets out of control) remain the most common type of IS problem
  – the prevalence of this problem increases with increases in systems size and complexity

USA Statistics

a survey (1997) by the US Government’s Accounting Agency found that:

  - less than 3% of the software that the US Government had paid for was actually being used as delivered
  - 75% of all development undertaken was either never completed or not used
    - 25% never delivered
    - 50% delivered but not used

Types of IS failure (Lyytinen & Hirschheim)

1. Correspondence failure
   - the system fails to meet the design objectives

2. Process failure
   - development problems lead to the project being unable to deliver a working system - project runs over budget in terms of cost or time

3. Interaction failure
   - if the system is heavily used it is a success - if it is hardly ever used it constitutes a failure

4. Expectation failure
   - the failure to meet the requirements of a key stakeholder group
Systems Disasters

- outright operational disasters should be added as a separate category
- real IS-based disasters are rare in practice but...
  - the potential for significant failure is hidden in many information systems which are operating successfully (Y2K provided a number of examples)

Analytical Frameworks

- the aim of research and analysis (group assignment) is to identify common themes and patterns of events
- a number of problem typologies have been proposed - just one is discussed in this lecture, but there are others
- the themes tend to recur, irrespective of the system type
- typologies or other types of analytical model facilitate analysis and presentation

The “LASCAD” Disaster

- the failure on which your case study was based
- an attempt was made at the inquiry (and the claim was repeated on a film made about it) to blame the disaster on a technical failure - in effect a queue became full and could not be cleared
  - it can be understood as typical of the tendency to look for technical causes of disasters, but was extremely short-sighted in that case
  - in normal circumstances, such a problem would have been discovered during testing, or at worst be dealt with by contingency arrangements
Technical Explanations

- technical explanations of failure very occasionally do tell the whole story
  - sometimes a bug may be buried deep within a system
  - eg - the Therac-25 example could be seen as a case in point
- but the LASCAD error related to a basic set of operations, and when that happens the root cause of the problem lies elsewhere (ref the Exxon gas explosion at Longford a couple of years back, or the Auckland power failure, or....)

Factors “encouraging” IS Failures

- cost-cutting and efficiency pressures
- the pace of technological change/pressure to respond
- interdependencies between different systems
- the costs of quality
- skills shortages
- constant stress
  - these are all risks which can spawn complex relationships

a Possible Typology of Problem Factors

- project
  - special characteristics: interest, significance
- psychological
  - pride, responsibility, don’t quit
- social
  - competition, prestige, justification
- organizational (structural)
  - funding, empires, weak management
  - typology appropriate for the analysis of runaway projects
Social/Technical System Matrix

- the matrix is based on the assumption that systems have two key dimensions
  - social dimension
  - technological dimension
- each of the matrix slots is a potential source for problems...

LAS - Matrix

- 4 x 4 possibilities for consideration
  - eg - theories in use/technology
- helps to classify one specific problem
  - one “theory in use” (belief/attitude about the system) was that the size of the LAS mandated a new system (no package)
  - but this was inconsistent with the implied “quick and dirty” technical approach
Structural Factors

• my own view is that IS/IT activities are particularly susceptible to “structural” problems

• the potential for a subsequent failure is inherent in the way in which the initial arrangements are made: critical are
  – authority structures
  – power relationships (not the same)
  – structures of objectives
  – financial relationships

The Problem of Stress

• structural arrangements can induce excessive stress
  – systems development and installation is a highly stressful activity in any case
  – structural relationships can create stress that has no outlet (the problems are beyond managers’ control)