Software Failures

Massimo Felici
How Software Projects Fail

- Software appears, by its nature, to be difficult to engineer on a large scale. Nevertheless, there is an insatiable demand for sizeable, well-engineered software.

- We continue to be dogged by large numbers of project failures, on small and large projects. Many (most?) of these are due to mistakes in project management.

- In this lecture we discuss:
  - Examples of project failure on a large scale
  - Lessons that can be learned
Recall: Criteria for success

For the purposes of this course, a software project will be considered successful if:

- The software is delivered on schedule; AND
- Development costs were within budget; AND
- The software meets the needs of users (in scope and quality)
Conversely: Criteria for failure

A software project may be considered a failure if:

- The software is not delivered on schedule (or perhaps, at all); OR

- Development costs are not within budget; OR

- The software does not meet the needs of users
  (it has unsatisfactory scope OR unsatisfactory quality)

In Standish CHAOS report terms, this covers both projects they describe as failed and those they describe as challenged.
Resolution of Projects

CHAOS 2004
Survey Results

Resolution of Projects

- Challenged: 53%
- Succeeded: 29%
- Failed: 18%

Copyright © 2009 The Standish Group International, Inc.
FBI Virtual Case File

- The US Federal Bureau of Investigation has often been criticised for not sharing leads between agents and divisions.

- Just before the 2001 terrorist attacks, the FBI hired Science Applications International Corp (SAIC) to develop Virtual Case File software (VCF).

- VCF was designed to manage case files electronically, so that any agent with suitable permissions can find relevant information.

- Originally scheduled for completion in 2003.
FBI Virtual Case File

After repeated delays, a version was delivered in December 2004, but:

- Was about one tenth of originally promised
- Was eventually scrapped altogether
- Does not approach functionality of existing commercial packages
- Used as an extremely expensive prototype
- About $170 million wasted
FBI Virtual Case File

Apparent causes:

• Changing requirements (after the September 11 attacks)

• Ambitious project, run as an emergency fix

• 14 different managers over project lifetime

• Poor oversight of external contractor

• Not paying attention to new, better commercial products

• Hardware purchased already; waiting on software
Supply Chain Management

• **Background:** fierce competition between UK supermarkets. Sainsbury’s fighting Tesco in particular. Supply chain management key to competitiveness.

• **pre-2000:** Sainsbury’s had “mainframe-based warehouse management system (WMS)”; “server utilisation as low as 1%”; “400 different supply chain software applications”.

• **2000:** new CEO Sir Peter Davis authorised outsourcing IT to Accenture, aiming to get an “agile IT infrastructure built on an open, adaptive, scalable architecture with hardware and software systems that would give very high performance, strong data security, and low total cost of ownership.” Key supplier Sun.
Supply Chain Management

- **May 2004**: “The $1.8 billion overhaul is well under way, and will be completed in 2005.”; “The relationship with Accenture has worked so well that Sainsbury’s has chosen to extend its IT outsourcing contract for another three years, until 2010 — a move that should allow the retailer to net additional cost reductions of more than $230 million by 2007.”

- **July 2004**: Davis resigns — poor financial performance

- **October 2004**: New system unable to track stock correctly. Shops go short. Sainsbury’s recruits 3000 shelf stackers to handle crisis, writes off $260m IT spend, renegotiates contract with Accenture. Disputed blame. Accenture blames poor reliability of four fully automated depots, not covered by the agreement with Accenture; new Sainsbury CEO Justin King blames Accenture.

- **October 2005**: Outsourcing contract cancelled, IT brought back in house.
Supply Chain Management

Apparent causes of problems with the Accenture attempt:

- Weak outsourcing governance
- Loss of staff with knowledge about legacy systems
- Risky “big bang” approach
- Political in-fighting
- Generally poor business management

[Main source: Douglas Hayward, senior analyst at researcher Ovum, quoted in a silicon.com article]
Customer Database System

- In 1996 a US consumer group embarked on an 18-month, $1 million project to replace its customer database. The new system was delivered on time but didn’t work as promised, handling routine transactions smoothly but tripping over more complex ones.

- Within three weeks the database was shut down, transactions were processed by hand and a new team was brought in to rebuild the system.

- **Problems:**
  - The design team was over-optimistic in agreeing to requirements
  - Developers became fixated on deadlines, ignoring errors
Customer Tracking System

• In 1996 a San Francisco bank was poised to roll out an application for tracking customer calls. Reports provided by the new system would be going directly to the president of the bank and board of directors. An initial product demo seemed sluggish, but telephone banking division managers were assured by the designers that all was well.

• But the system crashed constantly, could not support multiple users at once and did not meet the bank’s security requirements. After three months the project was killed; resulting in a loss of approximately $200,000 in staff time and consulting fees.

• **Problems:**
  – The bank failed to check the quality of its contractors
  – Complicated reporting structure with no clear chain of command
  – Nobody “owned” the software
Payroll system

- The night before the launch of a new payroll system in a major US healthcare organisation, project managers hit problems. During a sample run, the off-the-shelf package began producing cheques for negative amounts, for sums larger than the top executive’s annual take-home pay, etc.

- Payroll was delivered on time for most employees but the incident damaged the relationship between information systems and the payroll and finance departments, and the programming manager resigned in disgrace.

- **Problems:**
  - The new system had not been tested under realistic conditions
  - Differences between old and new systems had not been explained (so $8.0 per hour was entered as $800 per hour)
  - “A lack of clear leadership was a problem from the beginning”
Distribution System

- Anticipating growth, a $100 million division of a $740 million manufacturing business earmarks $5 million for a new distribution and customer service system to replace its old one. The project is to take a year and a half to complete. Two years later, the CIO is sacked and a new executive brought in to save the project. Three months later, the system breaks down altogether.

- Nine months later, the CIO approached his boss, the CEO to tell him the project is a failure. “It was kind of like telling him a relative had died,” he recalls. “First he denied it, then he went through a grieving process, then he accepted it. It was just so much money for a division that size to wave in the wind.”

- **Problems:**
  - Wrong direction from the start
  - Inadequate software plan
  - Nobody “owned” the software
Critical Failure Factors

Warning signs of a project doomed to failure, or even disaster

- Organisation: hostile culture, poor reporting structures
- Management: over-commitment, political pressures
- Conduct of the project:
  - Initiation phase: technology focused, lure of leading edge, complexity underestimated
  - Analysis and design phase: poor consultation, design by committee, technical fix for management problem, poor procurement
  - Development phase: Staff turnover, poor competency, poor communication (e.g. split sites)
  - Implementation phase: receding deadlines, inadequate testing, inadequate user training
Summary

• SW development is inherently a risky process

• Many projects fail for the same reasons

• Unfortunately, hindsight is much clearer than foresight, but

• The risk of failure should be addressed from the very start
Required Readings

  http://dx.doi.org/10.1109/MSPEC.2005.1502528