Embracing Knowledge And Behaviour Management to Improve Performance of Software Intensive Projects

Ricardo Peculis, CSC, SEEC-UniSA
Derek Rogers, SEEC-UniSA
Peter Campbell, SEEC-UniSA
Topics

- Overview
- Software Intensive Projects
- Influence of Knowledge
- Influence of Behaviour
- Knowledge Management and KMP
- Behaviour Management and BMP
- Summary
- Conclusions
- Questions
Many reasons have been offered to explain why software projects fail:

- Lack of executive support,
- Lack of customer involvement,
- Lack of experienced project managers,
- Lack of clear business objectives,
- Lack of stable requirements,
- Lack of competent staff.
Overview

- Lack of Knowledge is yet another factor that causes projects to under perform,
  - which drives undesirable social behaviour,
  - that worsens the situation.

- Software projects fail because of:
  - Lack of technical knowledge to engineer the solution, and
  - Lack of management knowledge to recognise and plan for this deficiency.

- The performance of software intensive projects can be improved by the adoption of:
  - Knowledge Management, and
  - Behaviour Management.
Software Intensive Projects

- Apply the process of engineering to develop products highly dependent on software.

- Scale, life cycle and interdependency of components increase complexity.

- Complexity: our limitation to deal with systems with large number of interconnected and interdependent components.
Software Intensive Projects

- Software intensive systems are complex.

- To deal with complexity, software engineering processes adopt techniques that
  - Abstract complexity,
  - Hide details.

- Revels problems at late stages of the project.
Software Intensive Projects

- Software intensive systems are acquired through an acquisition process that procure a solution, highly dependent on software, to satisfy a need.

- The acquisition process can be represented by a set of products and transformations within the acquisition space.
Software Intensive Projects
The Acquisition Model

Ideal transformations between Domain Spaces
Software Intensive Projects
The Acquisition Model

Actual transformations between Domain Spaces
(Chinese Whisperer’s Effect)
Influence of Knowledge

- The development of software products needs knowledgeable, talented and motivated people.
  - Providers: formulate and solve the problem, and implement the solution.
  - Customers: communicate the need and recognise when the solution satisfies that need.
Influence of Knowledge

- Knowledge is incomplete and distributed.
  - No single individual or organisation knows everything.
  - Individuals and organisations are more knowledgeable of subjects that are pertinent to the domain where they operate.
  - Distributed knowledge creates different perceptions and mental models.
  - The number of organisations in the system increases the effects of distributed knowledge.
Influence of Behaviour

- Knowledge is important but not enough.
- People need to be motivated to do the job.
  - Learn new subjects;
  - Acquire new skills;
  - Cooperate;
  - Teaching, coaching and helping others.
- People have different behaviour styles:
  - Constructive;
  - Aggressive; or
  - Passive.
- Behaviour styles cause people to respond differently to situations presented to them.
Influence of Behaviour

- Software intensive projects are problem-solving groups.
- The objectives of problem-solving groups are
  - Maximise the quality of results.
  - Contribution of each member in the group.
  - Maximise the acceptance of the solution.
  - Motivation for carrying out the group’s decision.

(Hoffman, 1994)
Influence of Behaviour

- The effectiveness of problem-solving groups is affected by behaviour styles.
  - Constructive behaviour
    - Increases the quality of the solution;
    - Increases the acceptance of the solution.
  - Passive behaviour
    - Decreases the quality of the solution;
    - Decreases the acceptance of the solution.
  - Aggressive behaviour
    - Decreases the acceptance of the solution;
    - Does not impact on the quality of the solution.

(Cooke & Szumal, 1994)
Influence of Behaviour
The Influence of Stakeholders

- Senior Management
  - Capable of supporting or ending the project.
  - Influenced by stakeholders external to the project.

- Managers, senior engineers, natural leaders
  - Influence the success or failure of the project.
Influence of Behaviour
The Influence of Stakeholders

- Projects that are not performing and put interests are at risk
  - Create tension between Project Management and Engineering Management.
  - Caused by the difference in nature of management (physical) x engineering (functional) activities. (Aslaksen, 1996)

- Decisions are made within constrains.
  - Unlikely to address lack of knowledge;
  - Often produce undesirable social behaviour:
    - Decreases motivation;
    - Discourages learning and cooperation.
  - Worsens the situation.
Knowledge Management

- Process of creating, disseminating and applying knowledge.
Knowledge Management

Specific Knowledge Management Plan (KMP)

- Identify the knowledge required to execute the project.
  - Areas of knowledge.
  - Policies for selecting organisations and staff for the project.

- Identify the knowledge available and where it is located.
  - Requires a process for measuring intangible assets (knowledge, skills and experience).

- Identify the knowledge that is missing.
- Plan to harvest existing knowledge.
- Plan to acquire the knowledge that is not readily available.
Behaviour Management

- Behaviour is driven by organisational culture and impacts on the level of cooperation.
- Behaviour styles impact on the effectiveness of Knowledge Management (Balthazard & Cooke, 2004)
  - Constructive behaviour promotes cooperation and knowledge exchange;
  - Non-constructive behaviours undermine attempts of achieving KM goals.

- Objectives of Behaviour Management:
  - Maximise constructive behaviours;
  - Promote cooperation;
  - Maximise the effectiveness of KM.
Specific Behaviour Management Plan (BMP)

- Identify likely behaviours within the project’s social environment;
- Plan to maximise the effectiveness of the organisation:
  - Stakeholder Management and
  - Organisational Culture Management.
Behaviour Management

- Stakeholder Management
  - Identify people and organisations capable of influencing the project.
  - Develop strategies to create a culture of constructive behaviour.
  - Attempts to ascertain stakeholders interests and their likely behaviour.

- Stakeholder Management tool
  - Stakeholder circle. (Bourne & Walker, 2003)
Behaviour Management

- Organisational Culture Management
  - Assert organisational culture.
  - Develop and organisational culture that best fits the organisation’s objectives.

- Organisational and Behaviour Management tools
  - Organisational Culture Inventory (OCI)
  - Life Style Inventory (LSI)

Summary

- Software intensive projects are problem-solving groups, and success depends on
  - Knowledgeable, talented and motivated people;
  - The way mental models are created and communicated;
  - The way people interact and apply their knowledge and experience.

- Knowledge is distributed among people and organisations.
  - Learning and cooperation is of fundamental importance to transform distributed knowledge into collective knowledge.
  - Collective knowledge is an emergent property that should be addressed by socio-organisational design.
Summary

- Knowledge and behaviour influence each other.
  - Constructive behaviours promote learning and cooperation and increase motivation.
  - Lack of knowledge causes projects to underperform and creates undesirable non-constructive behaviour.
  - Non-constructive behaviours lower motivation to learn and cooperate, and move the project away from desirable outcomes.
Conclusion

- Embracing Knowledge and Behaviour Management is an important element of socio-organisational design of complex projects.

- Knowledge and Behaviour Management can improve performance and chances of success of software intensive projects.

- Effective results will emerge when the commitment to KM and BM is embraced by all organisations involved in the software intensive acquisition as a whole.
Questions?