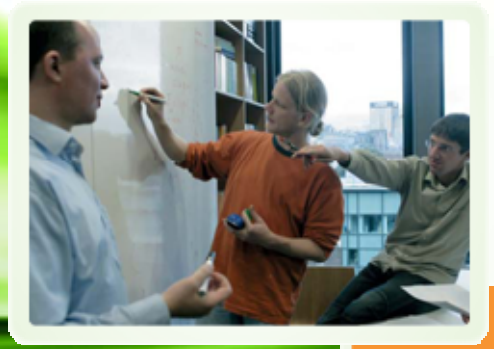


NICTA

From **imagination** to **impact**

Australian Government
Department of Broadband, Communications and the Digital Economy
Australian Research Council

NICTA Members

ANU UNSW Department of State and Regional Development

NICTA Partners

Victoria The Place To Be · The University of Melbourne · The University of Sydney · Queensland Government · Griffith University · QUT · The University Of Queensland

ICSP 2008, Leipzig

Macro-Processes Informing Micro-Processes: The Case of Software Project Performance

Dr. Paul Bannerman

Structure

- Motivation
- Process context
- Case of software development project performance
- Case example
- Implications
- Conclusions

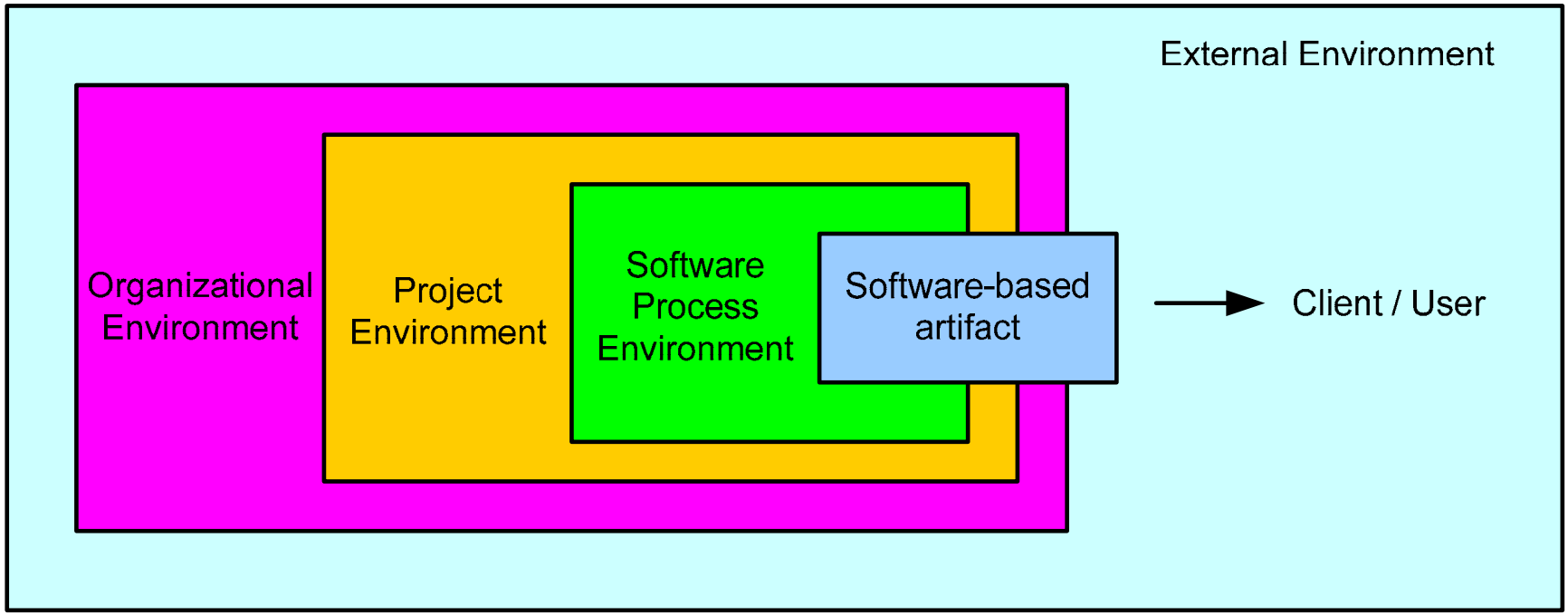
Motivation

Osterweil (2005):

- SP research approaches are complementary:
  Microprocess: internal structuring and workings
  Macroprocess: higher level behaviours
- Micro SP research explains macroprocess behaviours

This paper considers how macroprocess research (on software project performance) can inform microprocess research

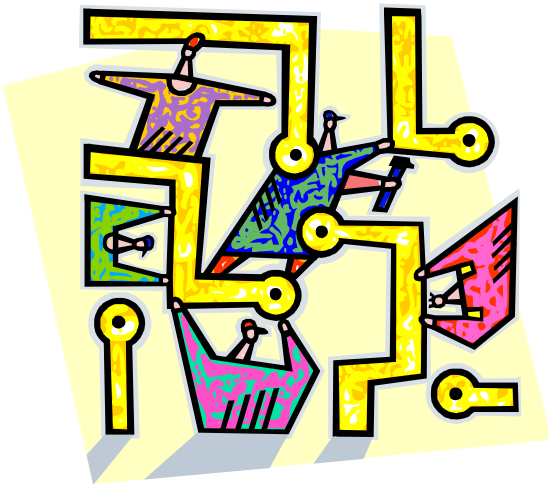
Process Context



Interdependent processes and environments

Case of Software Development Project Performance

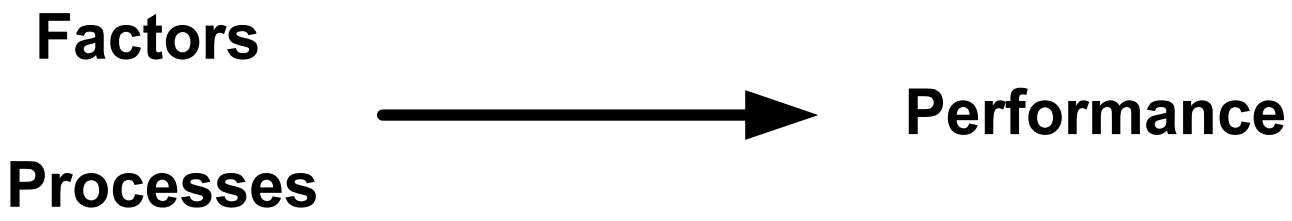
- Software macroprocess
- Why do software projects tend to fail?
- Capabilities-based explanation
- Implications for microprocess level



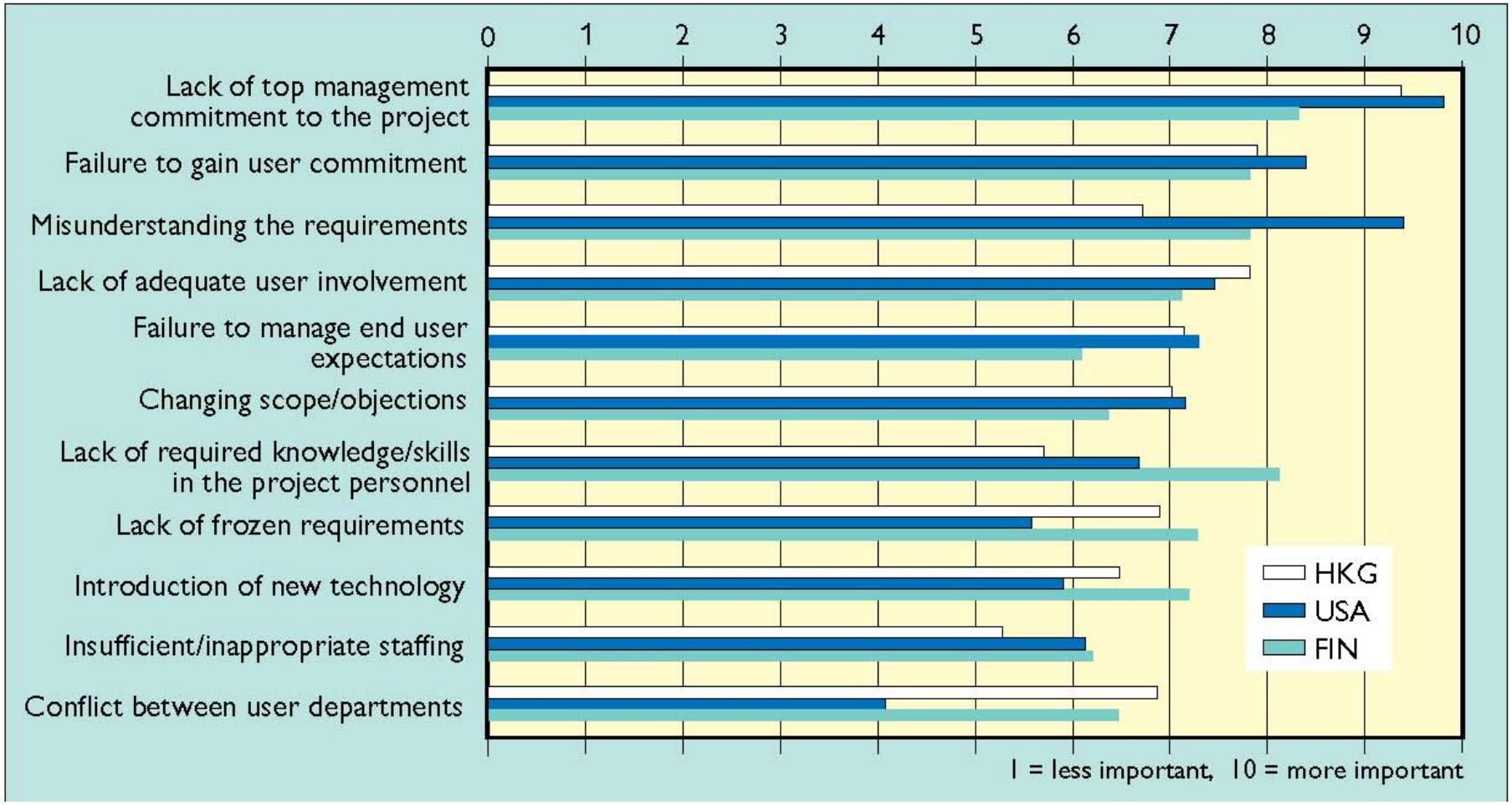
Why do Software Projects tend to Underperform?

Current explanations/solutions:

- Factors (e.g., success/failure/risk factors) – must take account of all CSFs for the project to succeed ("do the right things …")
- Processes (e.g., software/project/risk/change management processes) – must use best practice processes for the project to succeed ("… in the right ways")



Example: Software Projects Risk Factors



Keil et al., *A Framework for Identifying Software Project Risks*, CACM, 1998.

Example: Software Project Processes

- Project life cycle
- Project management methodology
- System development methodology
- Configuration mgnt
- Risk management
- Change management
- Benefits realisation

- Code & fix
- Waterfall
- Spiral
- Incremental
- Evolutionary
- Agile methods
- Object oriented methods

Limitations of Current Explanations

- Assume success is the ‘normal’ outcome (has no driver for failure)
- Assume that if you do the right things (factors) in the right ways (processes) then you will succeed
- Assume continuity - that more desirable factors / processes will improve your chance of success
- Assume that over time, you will get better (learn)

Empirical performance data does not support this view

Reframing the problem

Contrary to the assumptions of current approaches:

- Learning does not accumulate continuously
- Factors are not necessarily continuously additive
- Processes do not apply consistently in every context

Alternative assumptions:

- Data is correct (projects are more likely to fail than succeed)
- Improvement is not continuous (learning is discontinuous)

The world is fundamentally *discontinuous*:
success does not easily replicate

Capabilities-based explanation

Organisational capabilities:

- Performance is a function of firm resources/capabilities (‘resource-based view of the firm’)
- Distinctive capabilities can generate strategic value

Organisational learning:

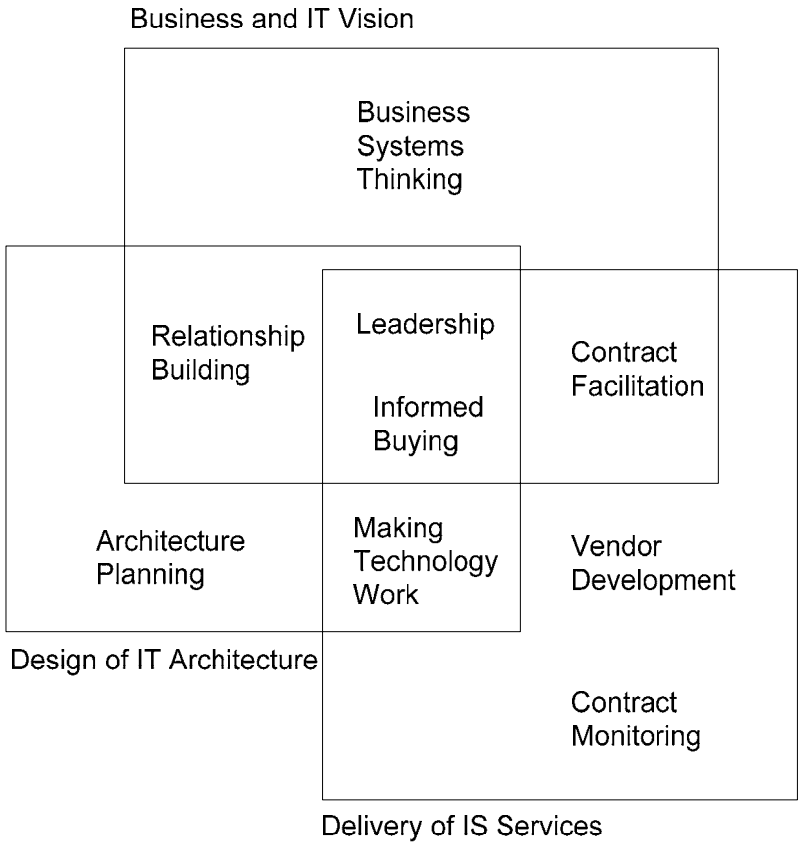
- ‘Learning by doing’ (from experience) is the generative mechanism of capabilities – continuous (‘single-loop learning’); discontinuous (‘double-loop learning’)

Organisational barriers:

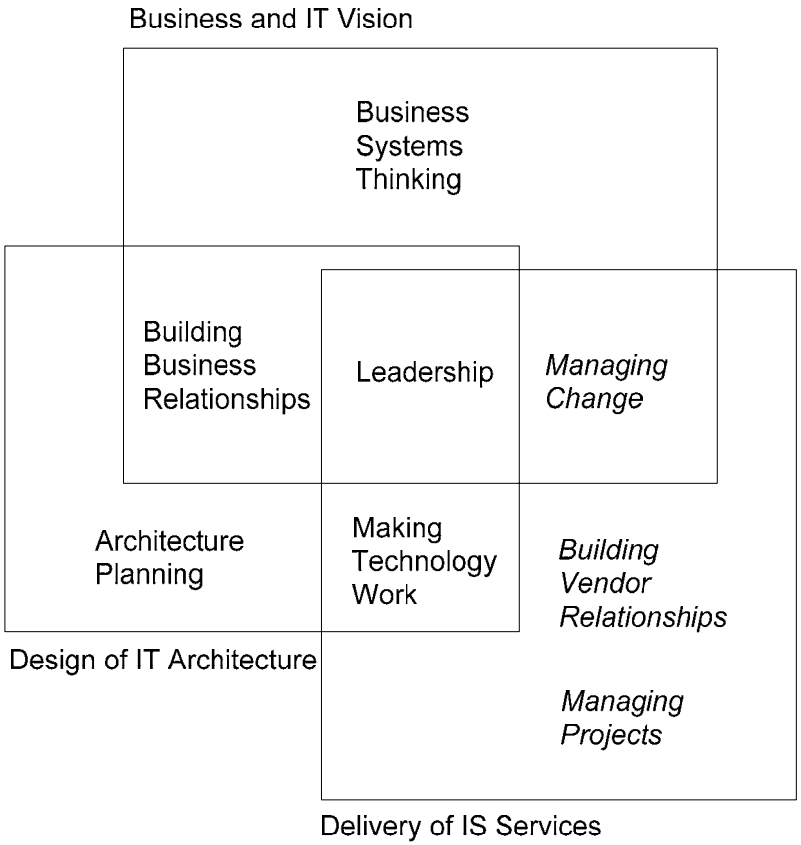
- Conditions that impede or block learning (continuous barriers)
- Conditions that make existing capabilities redundant or obsolete (discontinuous barriers)

Organisational Capabilities Example (1)

Outsourced Delivery

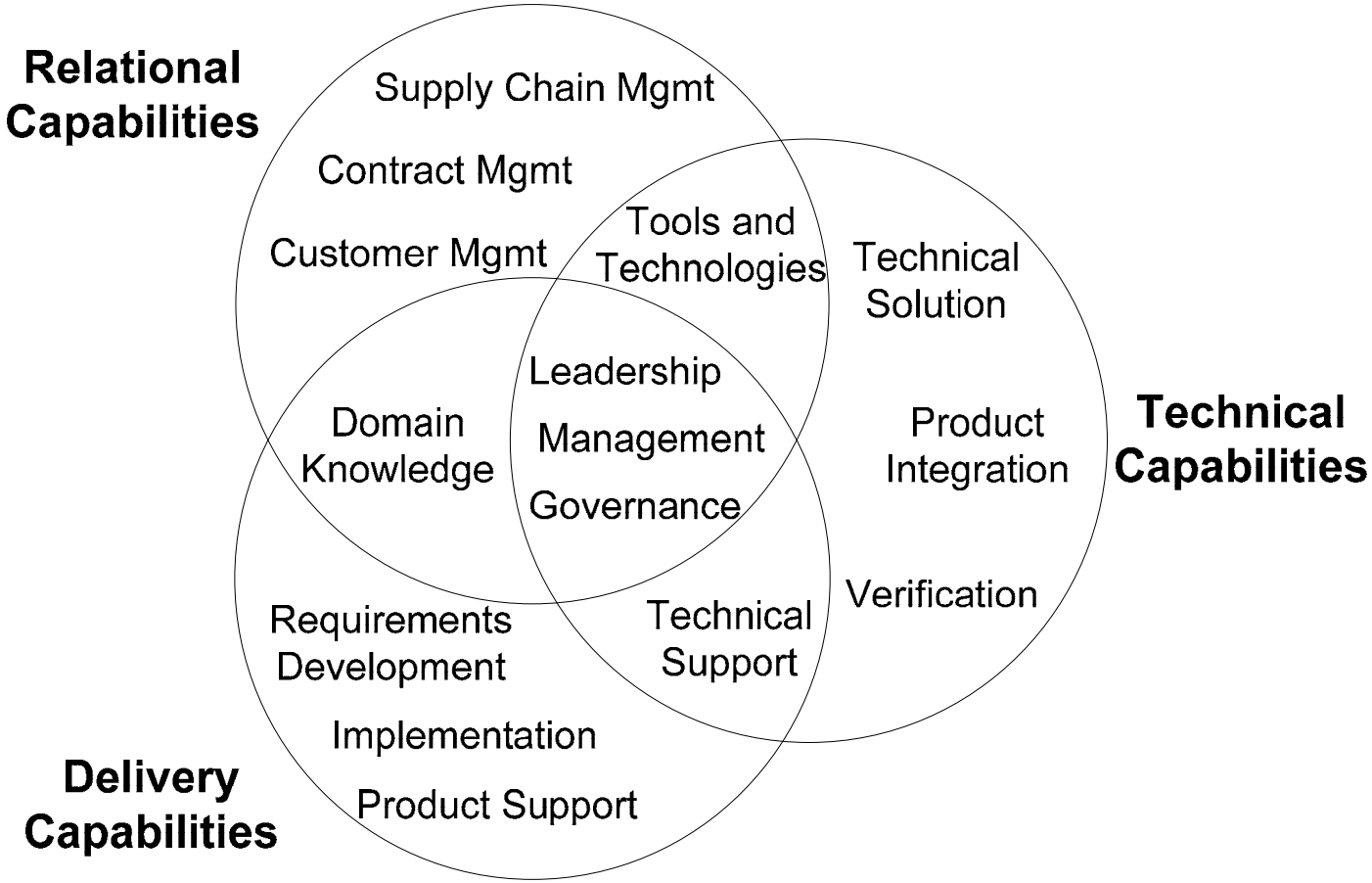


Internal & External Delivery



Feeny & Willcocks (1998)

Organisational Capabilities Example (2)



Bannerman & Staples (forthcoming)

Barrier Conditions Examples

Impede

- time compression diseconomies
- asset mass inefficiencies
- absorptive capacity
- learning disincentives
- certain organizational designs
- low aspiration levels

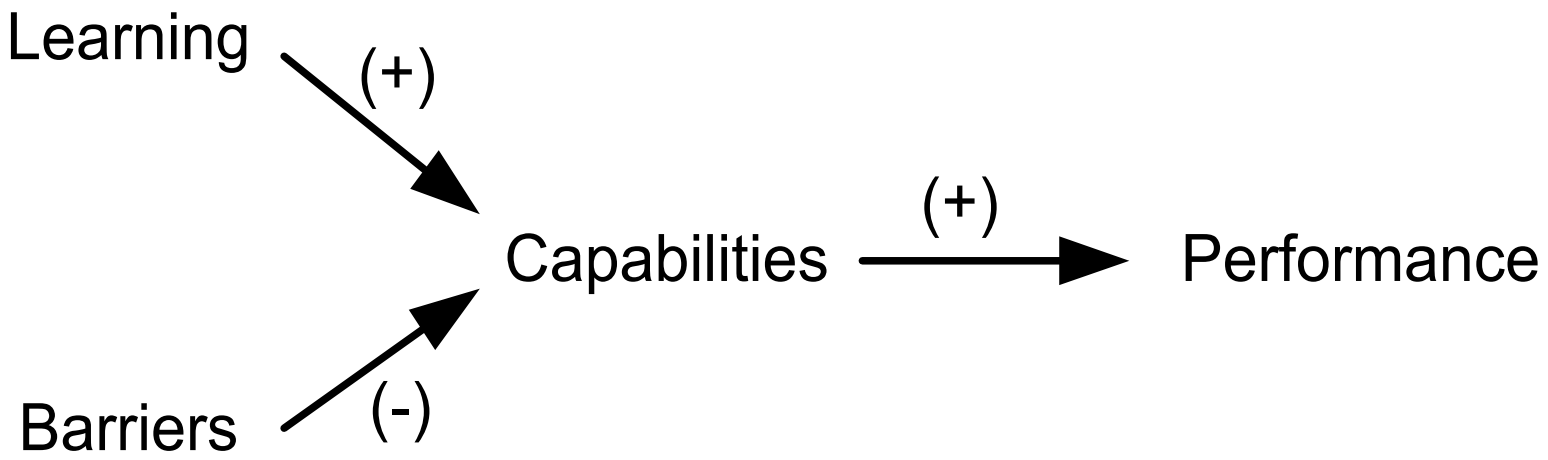
Block

- tacitness
- organizational inertia
- competency traps
- need for unlearning
- Interconnectedness
- causal ambiguity
- learning myopia
- focus diversion
- core rigidities
- complexity
- embeddedness
- ‘stickiness’
- unjustified theories-in-use
- managerial cognition
- certain characteristics of projects

Negate

- newness
- technological discontinuities
- ‘architectural’ innovations
- radical management methods such as *tabula rasa* business process reengineering
- staff loss through turnover, downsizing or outsourcing
- organizational forgetting
- asset erosion

Capability-based Performance



Performance is the contested outcome of drivers for success (organisational learning) and drivers for failure (organisational liabilities) on organisational capabilities

Case Study: Australian DMV (1)

- Major system development project
- Longitudinal case study spanning 1989 to 2001
- Initial development late, over budget, under specification
- Won award for excellence (because the major business objective were eventually achieved)
- Subsequent performance was inconsistent and unremarkable

Why?

Case Study: Australian DMV (2)

Capabilities by phase

IS Core Capability	Status Phase 1	Status Phase 2	Status Phase 3	Status Phase 4	Net Change	Summary Explanation
Building Vendor Relationships	L	L	M	M	+	Learning & capability development restricted to single vendor
Managing Projects	M	M	M	M	None	Not seen as a specialist skill; methods remained deficient
Making Technology Work	M	H	M	M	+/-	Challenged by constant change; mostly insourced contractor skill
Business Systems Thinking	L	M	M	L	+/-	Focus on technical tasks; knowledge limited to team leaders
Building Business Relationships	L	M	M	L	+/-	Focus on technical tasks; goodwill through strong role separation
Leadership	H	M	M	M	-	Original CEO/CIO vision, synergy and alignment lost
Architecture Planning	M	M	L	L	-	Initial skills diminished through outsourcing and focus diversion
Managing Change	M	L	L	L	-	Not seen as IT responsibility; focus on technical change

Case Study: Australian DMV (3)

Capabilities summary by phase

	Phase 1	Phase 2	Phase 3	Phase 4
Low	3	2	2	4
Medium	4	5	6	4
High	1	1	0	0

	Number	Direction
Opportunities for improvement	*24*	
Improvements	4	+
Decreases	6	-
No changes	14	N/A
Net changes	*-2*	

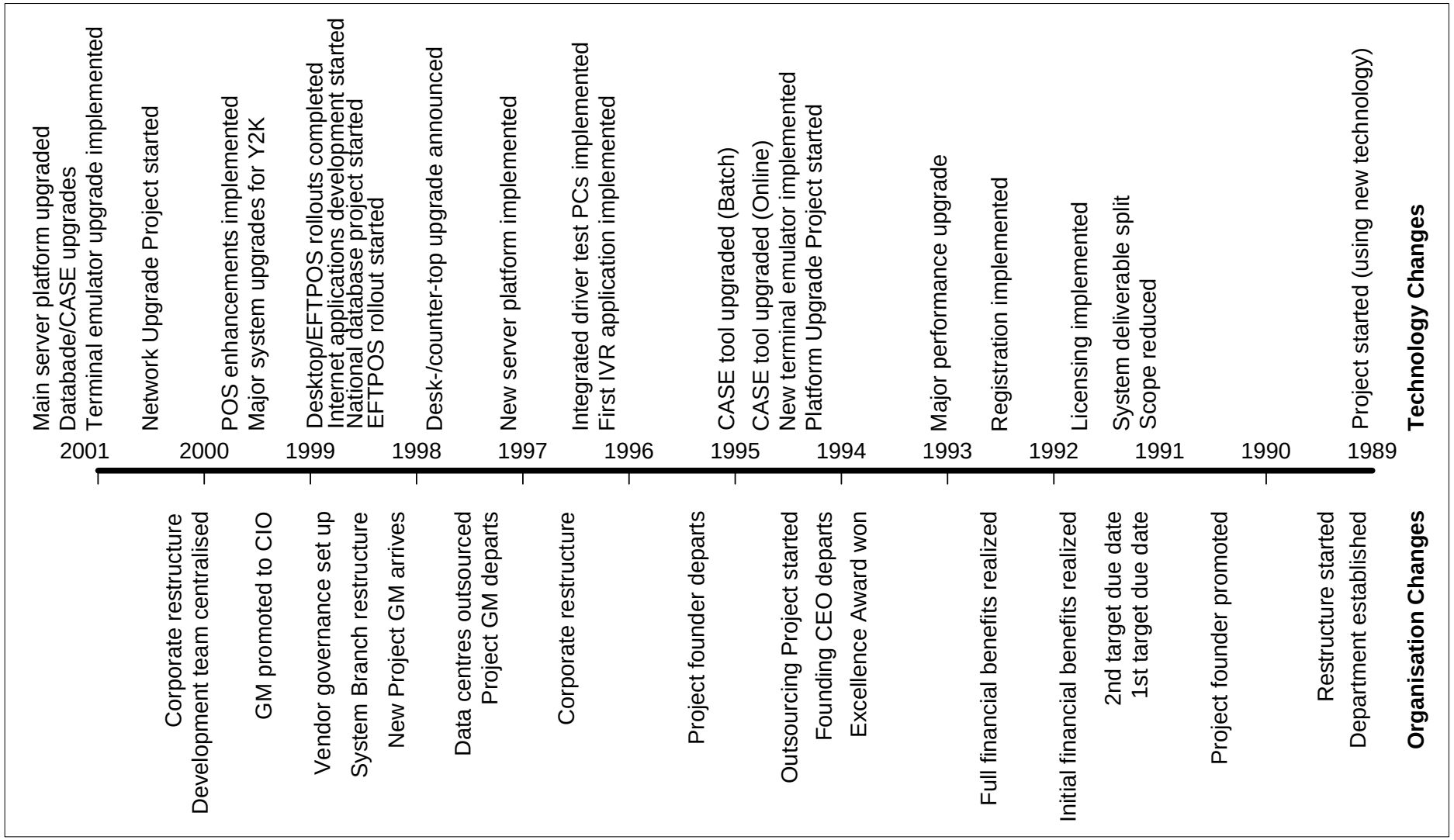
Case Study: Australian DMV (4)

Barriers by phase

Barrier Condition	Phase 1	Phase 2	Phase 3	Phase 4	Total
Newness Discontinuity	7	7	6	6	26
Influence of Past Practices	4	4	5	5	18
Constrained Focus (/Focus Diversion)	7	6	7	7	27
Capability Erosion	-	3	4	2	9
Knowledge Transfer Barrier	-	2	6	6	14
Limited Critical Mass	5	3	2	3	13
Time Compression Barrier	✓	✓	✓	✓	
Weak Learning Culture	✓	✓	✓	✓	
Barrier Condition/Capability Impacts	23	25	30	29	107

Over 80% of capabilities were impacted by newness discontinuities; overall, 56% of all capabilities experienced negative impacts from one or more barrier condition

Case Study: Australian DMV (5)



Case Study: Australian DMV (6)

Summary of Findings:

- No cumulative improvement was found in IS core capabilities
- Capability accumulation was found to be associated with barrier conditions
- Any organisational learning and capability development that occurred was offset by the effects of barrier conditions, resulting in a cumulative net competency liability rather than accumulated learning and capabilities
- The DMV was in no better condition after the project to improve its likelihood of success for any subsequent project

Implications of this Explanation

- Knowing what is needed for success, and how to do it, is insufficient without also having the enabling capabilities
- Performance is a function of fostering organisational learning and mitigating organisational liabilities
- ‘Best practice’ is necessary but not sufficient for success
- Some organisations may never be able to develop the competencies needed to ensure the success

Other ‘Classic’ Case Examples

- London Ambulance Service
  Computer Aided Despatch System
  - failed – 1990; 1992
  - succeeded – 1996

- Bank of America
  - ERMA – strategic success
  - MasterNet - failed

- American Airlines
  - SABRE – strategic success
  - CONFIRM – failed

Implications for Software Microprocesses (1)

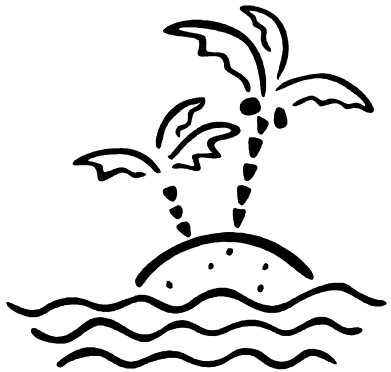
- Factor and process research may not be sufficient to adequately explain the behaviour of the phenomenon (SP effectiveness may reflect underlying competencies)
- Software process can be an organisational capability (an undifferentiated resource or a distinctive competency)
- Learning does not necessarily accumulate continuously (most SPI models assume this)

Implications for Software Microprocesses (2)

- Macro and micro software processes are symbiotic (SQ is ultimately determined at the macro level but is substantially created at the micro level)
- SP/artifact ‘success’ may be determined by unrelated factors at the macroprocess (software project) level (need to understand and manage the whole context that can impact software process outcomes)
- Intangibles contribute to performance outcomes as well as explicit process capabilities (need mechanisms to capture capabilities in the software process domain)

Conclusions

- Contextual awareness is important in software processes
- Microprocesses and macroprocesses can explain and influence each other
- Capabilities-based theory can explain the performance of software projects
- Neither process environment is an island



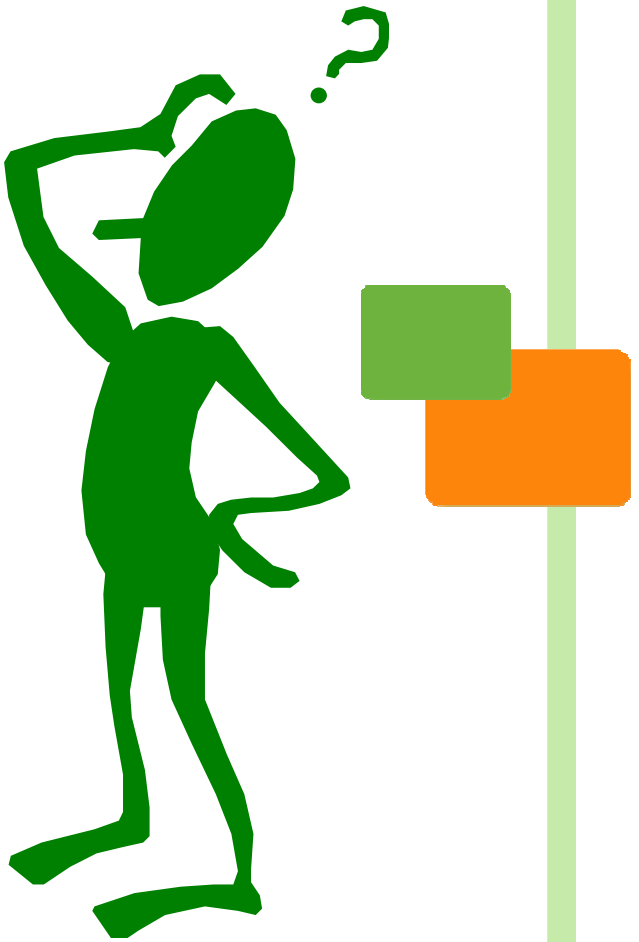
Questions?

NICTA

Thank you

Paul Bannerman

paul.bannerman@nicta.com.au



NICTA

From **imagination** to **impact**