

“Technochange and counter-design”

Bridging User Involvement in IT Architecture (BUITA)

ICM1

Magnus R. P. Hansen



Quick Recap of formative evaluation

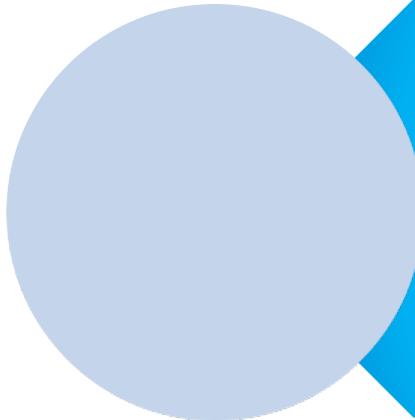


Technochange

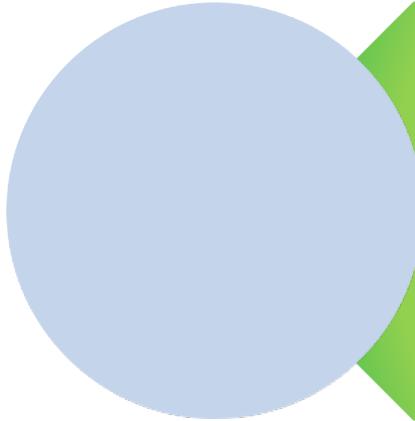


Implementation and change management

Articles:



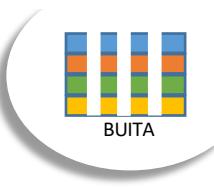
Markus, M. (2004). Technochange management: Using IT to drive organizational change. *Journal of Information Technology*, 19(1), 4-20.



Keen, P. G. W. (1981). Information systems and organizational change. *Communications of the ACM*, 24(1), 24-33.

Learning goals

- Know the differences between *technical implementation, organisational implementation through the concepts of IT projects, organisational change and technochange*
- Know the various strategies using systems development strategies such as *technochange prototyping, phased implementation evolutionary development, and pilot implementation*
- Know the central challenges of *counter-implementation*



Markus (2004)
Keen (1981)

Implementations in organisations

What the heck do we mean by “implementation” anyway?

Actors

- By whom?
- To whom?
- Of what?
- In what context?



Concepts

- Implementation
 - Short (when coding is finished)
 - Long phase (implementation preparations start when project starts)
- Deployment
 - Of hardware and software
- Activities
 - Training, Communication, Support
- Adoption
 - By decision stakeholders
 - By users

5 minutes of buzz

- For the two texts for today:
 - Which concepts and ideas do you expect and want to have cleared up?



Views on engaging organizations

Down and out (top down)

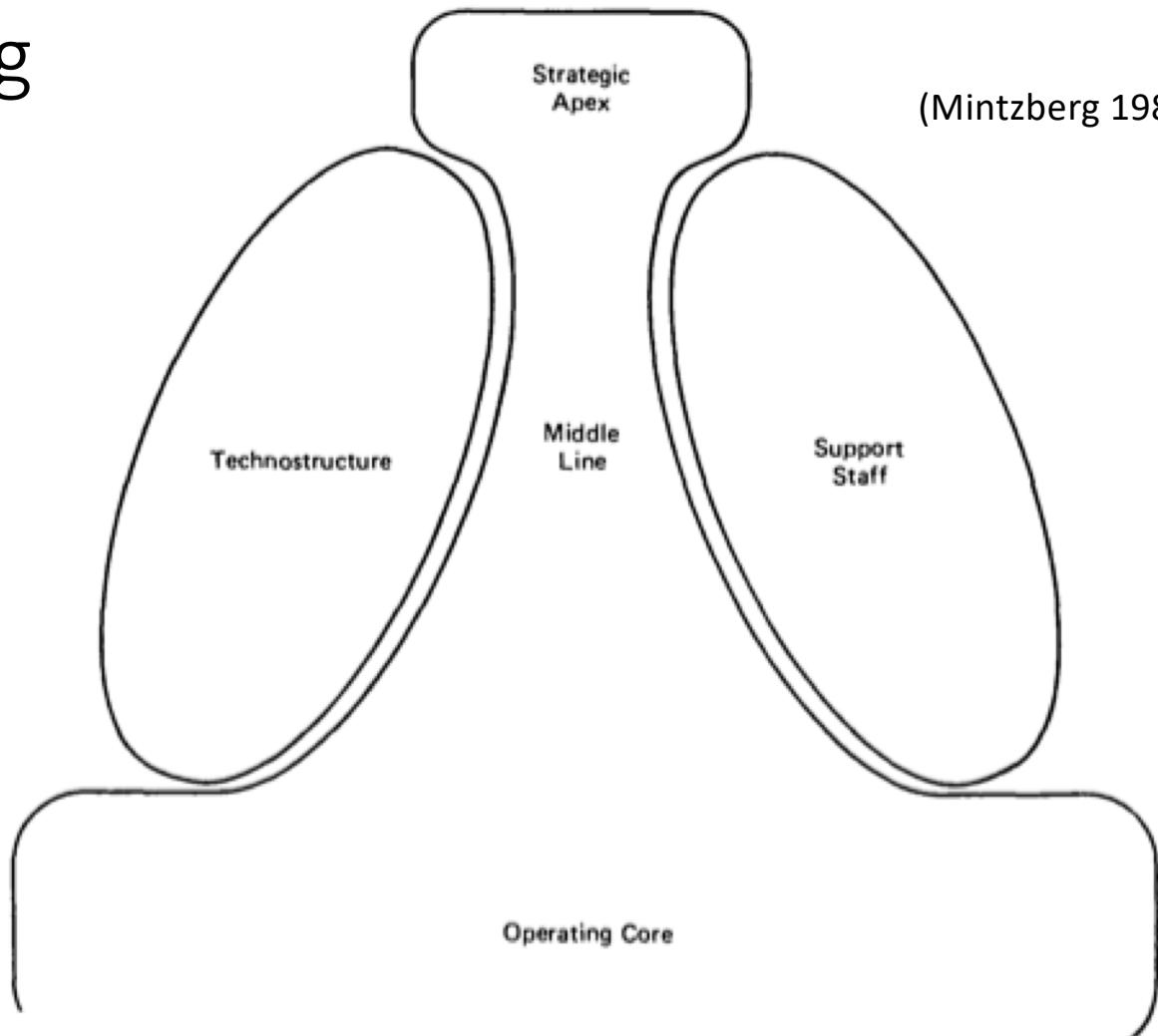
- Fast decision
- Assumes common purpose
- Lengthy design stage

Up and in (bottom up)

- Incremental change
- No limits in scope
- Requires consensus
- Lengthy decision stage

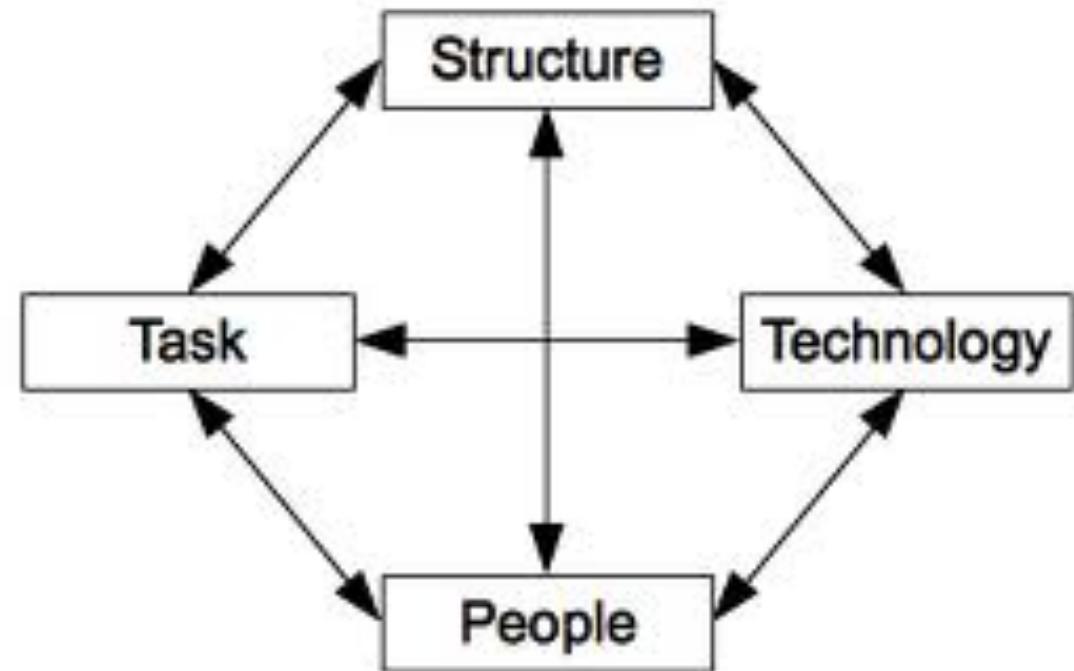
(Keen 1981)

(Mintzberg 1980)



Leavitt's Diamond model:

- All elements impact eachother

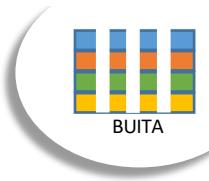


(Keen, 1981)

(Keen, 1981)

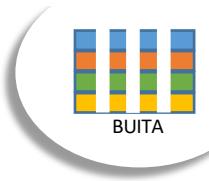
Decision processes

- Economic man
 - All alternatives are and can be laid out
 - We choose the best to find the "correct" means to an end
 - Maximizing
 - Administrative man
 - We want only a small list of alternatives
 - We want to simplify the situations as much as possible
 - Satisficing
 - Garbage can
 - We do what we can within the time and resources given to us
 - Reality prioritizes for us
- 
- What we *should strive for* and how we think we think
- What we *seem to prefer*
- What we *actually seem to do*
(read more in Simonsen, 1994)



Group task

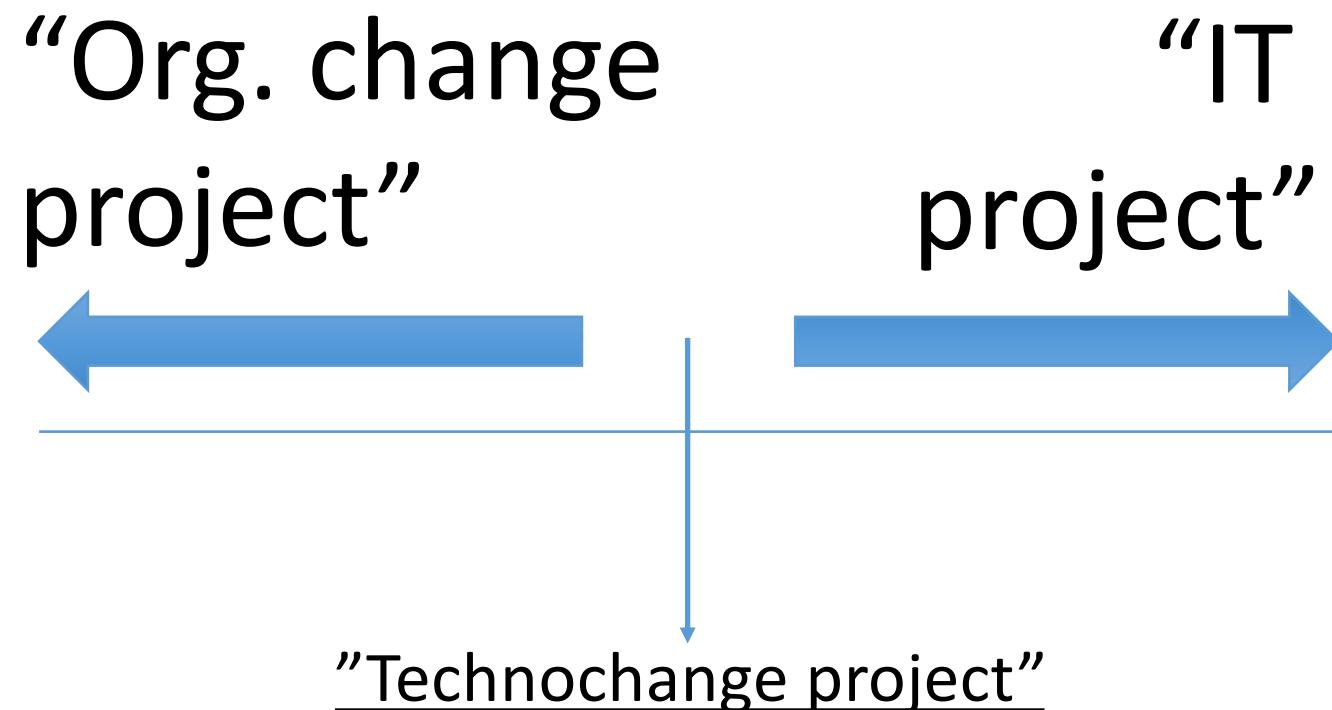
- How is your case organization structured?
- How does this impact



Technochange

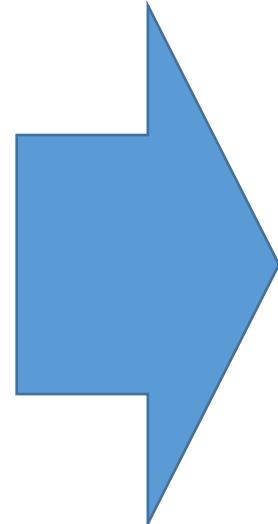
Markus (2004)

A continuum



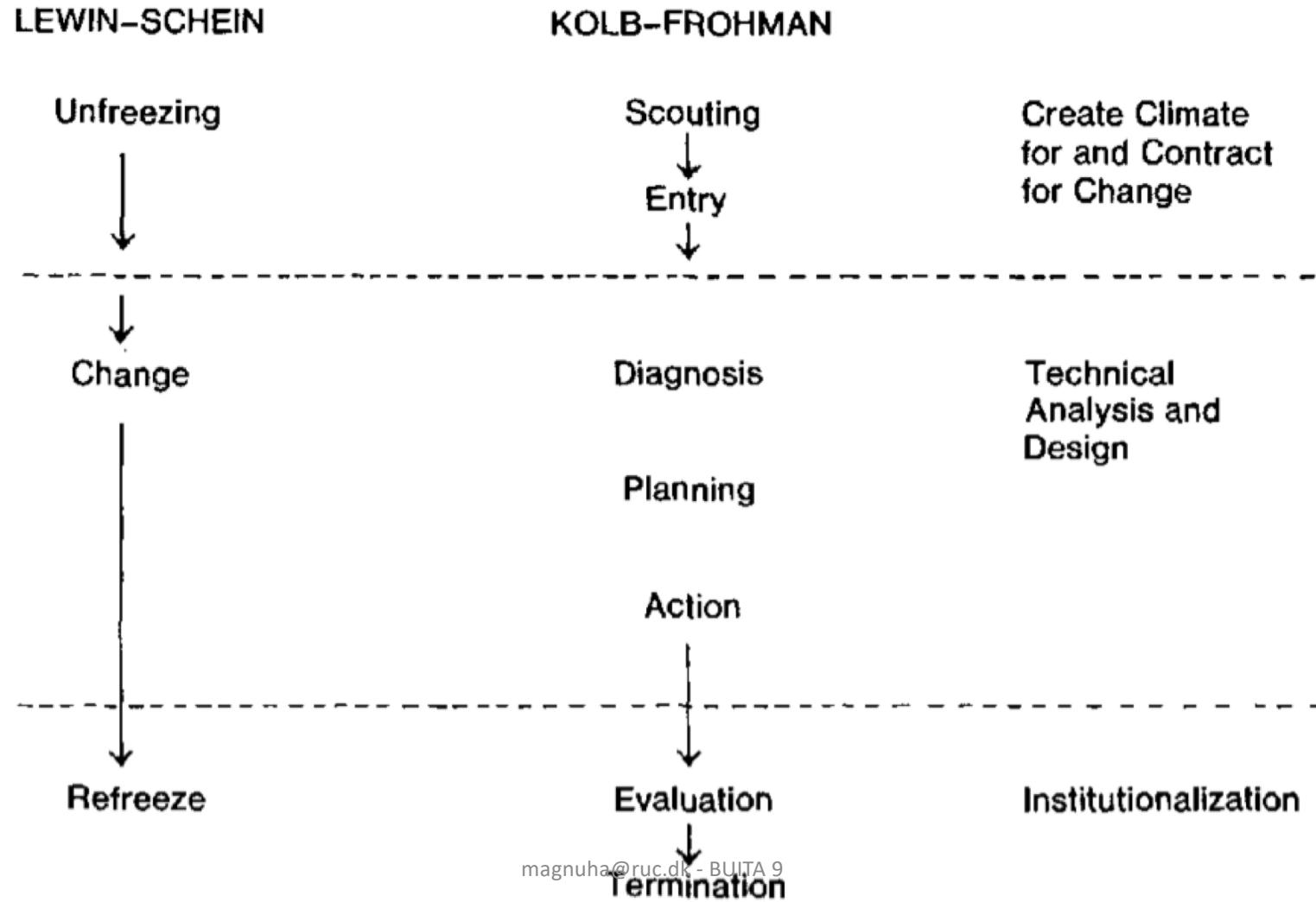
Requirements for an organizational change project

1. Sense of urgency
2. Guiding coalition
3. Create a vision
4. Communicate vision
5. Empower and train to act on vision
6. Short-term wins
7. Consolidate improvements
8. Institutionalize new approaches

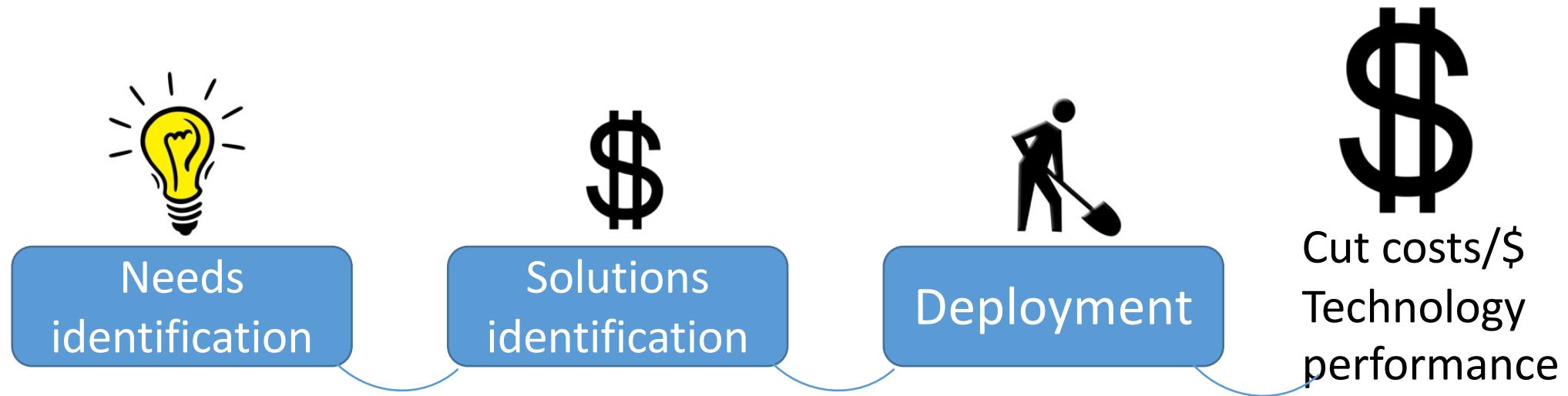


- Organizational performance
- Management performance
- Culture change

- Kotter, 1995



An IT project



However!

- "Some organisational changes *cannot* happen without IT" – p .6
- "Starting major cross-functional changes without an IT focus does not work in many organizational cultures." – p. 6



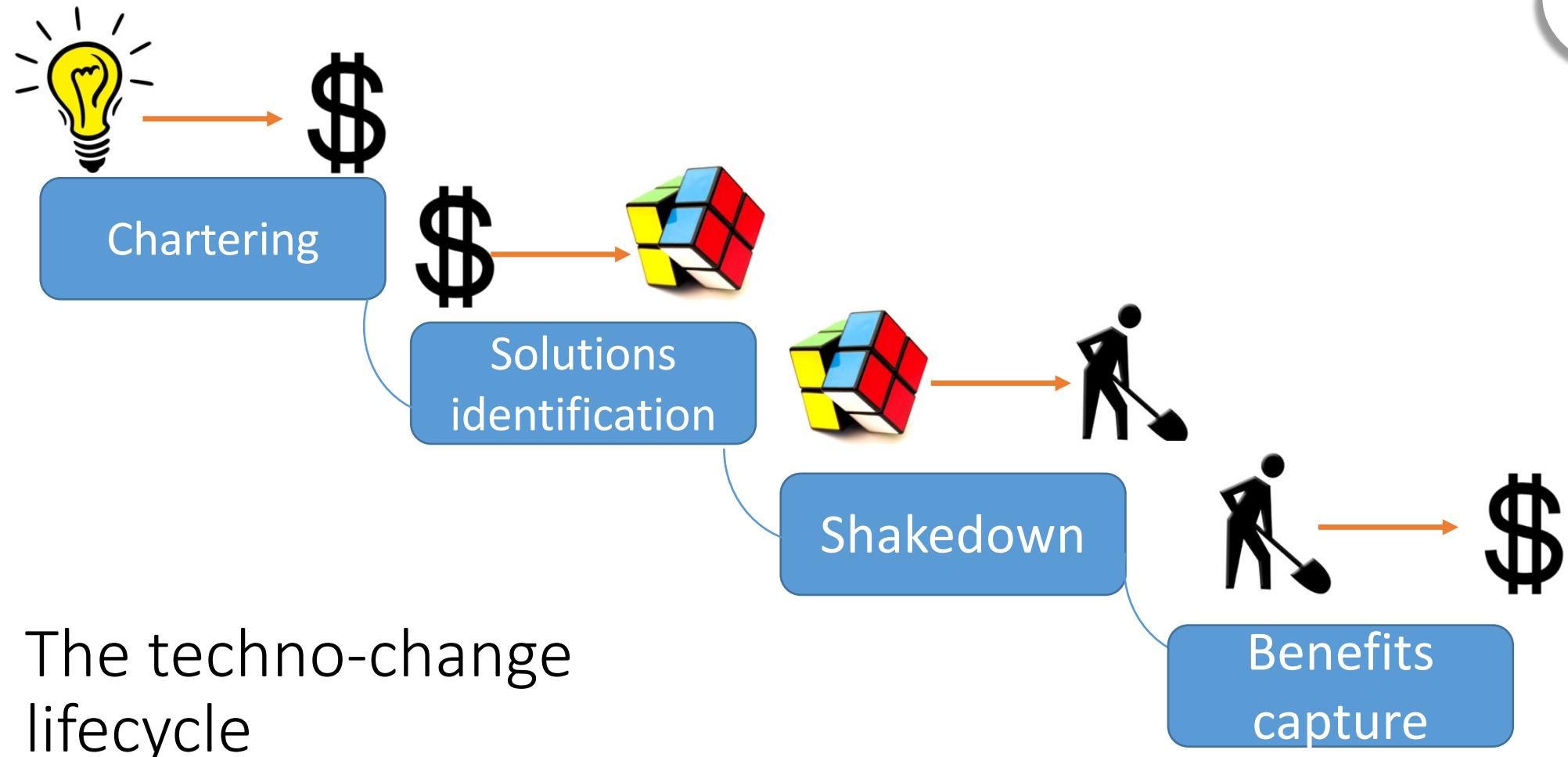
- "If you automate a bad business process, you get a faster, more expensive, bad business process." - p. 13
- "**Don't automate, obliterate**" – Hammer, p. 1, *Reengineering work, 1990*

Process	Product	Business result
Good	Good	Good
Poor	Good	Reject
Good	Poor	Poor
Poor	Poor	Reject

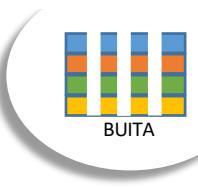
Defining *technochange*



- *Technochange* is fundamentally different from:
 - IT projects: improving technical performance
 - "What we do it *with*"
 - Organisational change programs: improving only processes
 - "*How* we do it"
- Technochange is the correct term
 - When IT impacts **users**
 - When you do **strategic use** of IT implementations
 - When you need to **bring people together**



The techno-change
lifecycle



In the “chartering” and “shakedown” phase:

1. Assess readiness for change
2. Train the users
3. Initiate cultural change
4. Redesign jobs, org. structures and existing tools
5. Find new ways to reward people
6. Involve users

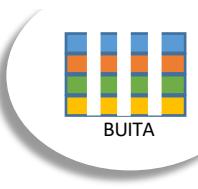
A supply chain management example

- Problem: difficulty to anticipate when inventory runs low.
- Solutions:
 - A. buy software package to improve forecasting of missing inventory
 - B. give suppliers online access to sales and forecast data for easier anticipation
 - C. require suppliers to take stock of supplier inventory, own inventory and point-of-sale data



Group discussion – defining your architecture project

- Where on the IT spectrum does your project exist?
- What kind of change are you driving towards?
 - Organisational
 - Performance indicators
 - Both?

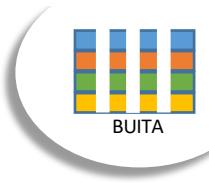


Technochange prototyping when impact is:

- More than a few people or groups
- On people outside IS department
- On people outside organisation
- On work tasks rather than work tools
- Expensive
- Revolutionary and not evolutionary

Technochange prototyping when:

1. Organisations is highly resistant to change
2. Technology is unproven
3. Technology is monolithic and indivisible
4. Organisational change is trialable



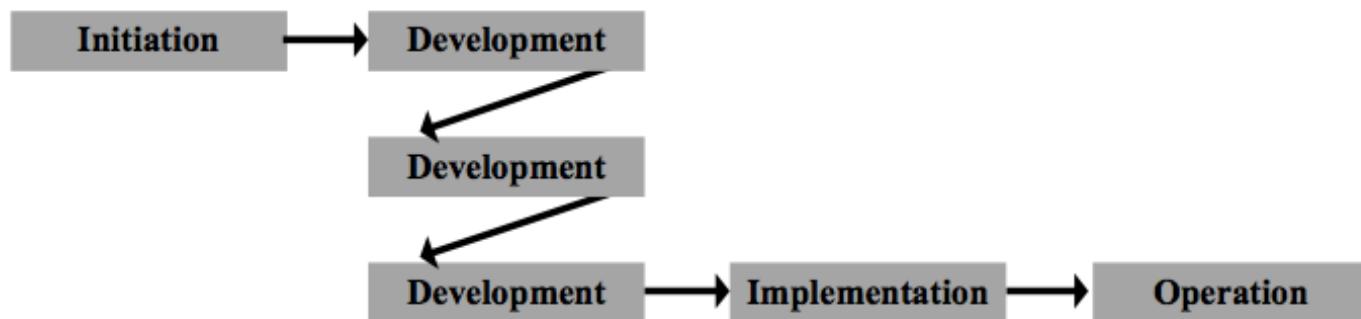
Types of development and implementation cycles

Detailed specifications - classic waterfall



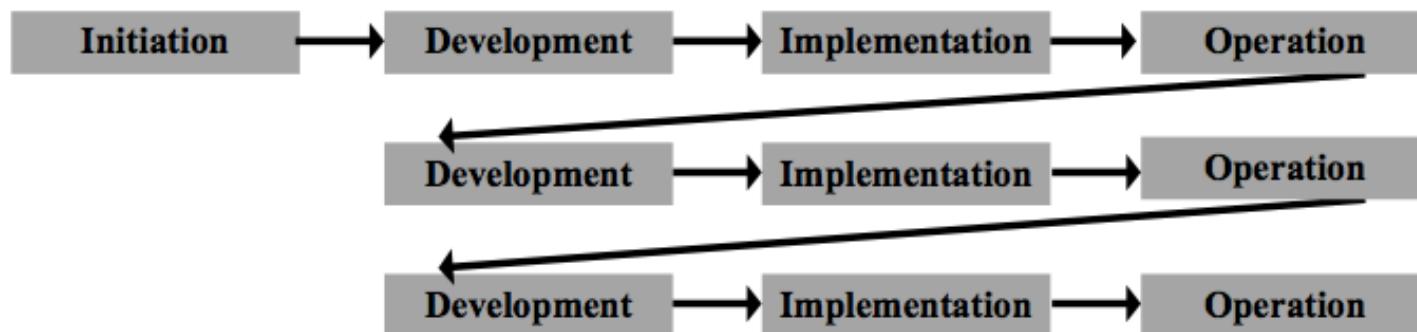
(Alter, 2001)

Iterative prototyping



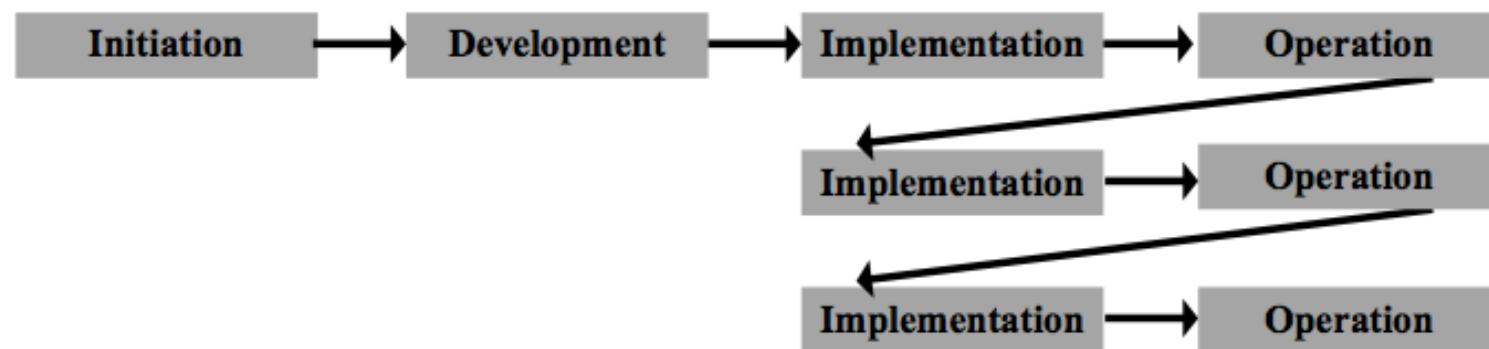
(Alter, 2001)

Evolutionary development



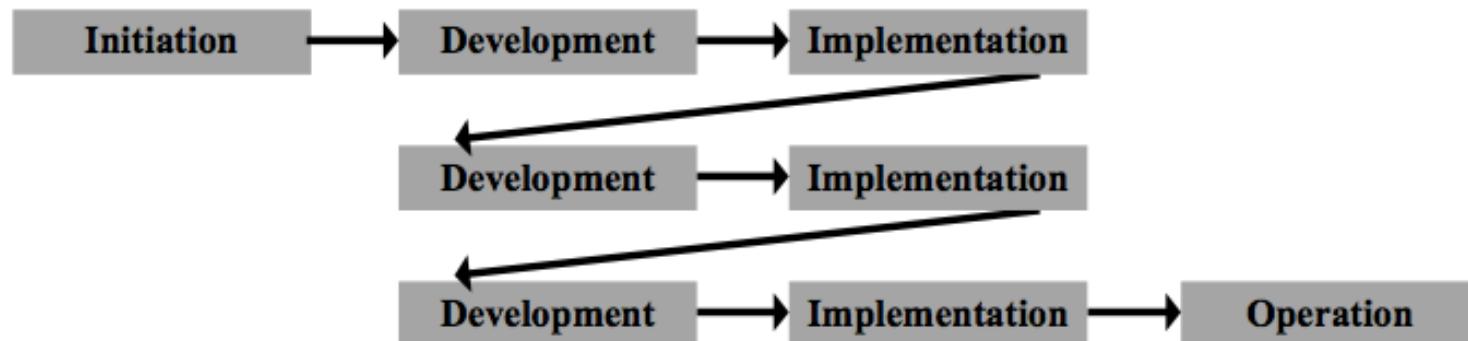
(Alter, 2001)

Phased implementation



(Alter, 2001)

Pilot implementation



(Hertzum et al., 2012)

When to do what?

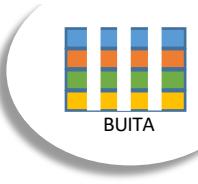
1. Detailed specifications - classic waterfall
2. Iterative prototyping
3. Evolutionary development
4. Phased implementation
5. Pilot implementation

When to do what?

- 1. Detailed specifications - classic waterfall
 - 2. Iterative prototyping
 - 3. Evolutionary development
 - 4. Phased implementation
 - 5. Pilot implementation
-
- 1. IT projects with stable environments
 - 2. Technochange on a large scale
 - 3. When doing IT projects with stable but reconfigurable technologies
 - 4. When doing IT projects in organisations with many different departments
 - 5. Technochange with a strong anti-failure and experimentally engaged org. culture, unstable environments

Issues and concerns with technochange

- Time and distance problems
- Exported problems
 - Passed on to next phase, politically powerless to fix the issues
- Technochange failure → culture of failure
- Unintended consequences



Group task - plan ahead of your scope

- Which strategy/strategies would you use for your architecture project?
- Show how the overall plan goes beyond your existing scope (use a flip over)

PHASE

1. Project establishment

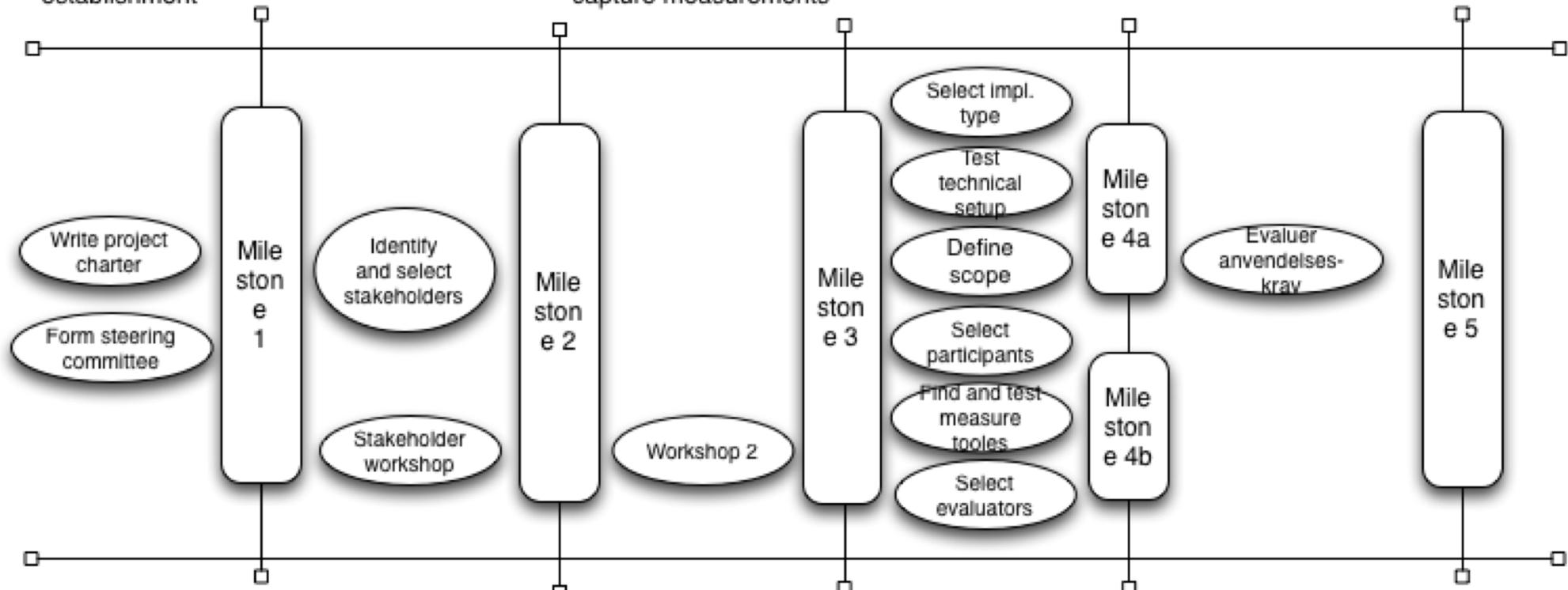
2. Specify benefits

3. Specify benefit capture measurements

4. Plan implementation

5. Perform implementation

PROCESS



PRODUCTS

1. Signed project charter, stakeholder analysis

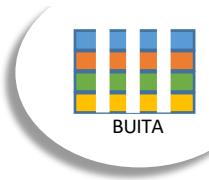
2. List of meta-requirements with causal links to believed benefits

3. Prioritized list of benefits with tools for measurement

4a. Technical setup operational
4b. Architecture, roles and responsibility document

5. Evaluation report:
- benefit results
- improvement areas, technical and organisational

EXAMPLE OF PILOT IMPLEMENTATION TIMELINE

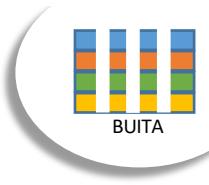


Keen, 1981

Social inertia

Counterimplementation

Countercounterimplementation



Social inertia

“No matter what you do, nothing seem to happen”

Social inertia reasons:



1. "Information is only a small component of organizational decision processes;
2. Human information-processing is experiential and relies on simplification;

Social inertia reasons:



3. Organizations are complex and change is incremental and evolutionary; large steps are avoided, even resisted;
4. Data are not merely an intellectual commodity but a political resource, whose redistribution through new information systems affects the interests of particular groups.” – p. 24

Counter-implementation games

- Diverting resources
- Deflecting goals
- Dissipating energies

How do you yourself
understand these
“games” (p. 29?

Diverting resources

1. "Easy money" → "Get a little more than we give back"
2. "Budget" → "We never turn down money"
3. "Easy Life" → "Make sure we're in charge, take it slow"
4. "Pork Barrel" → "Grab it while you can"



Deflecting goals

1. "Piling on" → "Let's do it right! We have to make sure"
2. "Up for grabs" → "If they don't know, we'll take over"
3. "Keep the peace" → "We have to work closely with marketing"



Dissipating energies

1. "Tenacity" → "No. We're not happy about..."
2. "Territory" → "This is OUR job. We should run the project."
3. "Not our problem" → "Marketing should handle this."
4. "Odd man out" → "We'll be happy to provide inputs, but..."
5. "Reputation" → "I want an integrated-online-realtime-databasemanagement-distributed-cloudbased-planning system. My system will..."



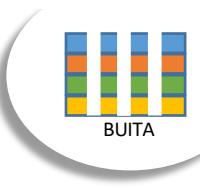
Strategies

Counterimplementation strategy:

1. Lay low;
2. Rely on inertia;
3. Keep the project complex, hard to coordinate, and vaguely defined;
4. Minimize the implementers' legitimacy and influence;
5. Exploit their lack of inside knowledge

Countercounter-implementation strategy:

1. Make sure you have a contract for change;
2. Seek out resistance and treat it as a signal to be responded to;
3. Rely on face-to-face contracts;
4. Become an insider and work hard to build personal credibility;
5. Co-opt users early.



Scenario-writing as part of implementation planning

- Basic objectives - what are you trying to do?
- Dilemmas of administration – how will you deal with massive resistance?
- Games – what games will divert resources, deflect goals, dissipate energies?
- Delay – how much delay is expected and how can you use project management as negotiation?
- Fixing the game – can you build a coalition to fix the game?

(full table p. 30)

Group task:

- Fill out a scenario that answers the questions on p. 30
- Upload to moodle when done

Learning goals

- Know the differences between *technical implementation, organisational implementation through the concepts of IT projects, organisational change and technochange*
- Know the various strategies using systems development strategies such as *technochange prototyping, phased implementation evolutionary development, and pilot implementation*
- Know the central challenges of *counter-implementation*

References

- Alter, Steven. 2001. “Which Life Cycle, - Work System, Information System, or Software?” *Communications of the AIS* 7 (17): 1–53.