‘Change for the better’: An architectural challenge

Tom Graves
Principal Consultant, Tetradian
The challenge

What architectures do we need to ensure that every change will make things better?
Four key concerns

1. How do we identify ‘better’?
2. What structures would support ‘better’?
3. What discipline is needed for change?
4. How do we verify and improve?
1. How do we identify ‘better’?

*Requirement:*

“A totem-pole to unify the tribes”

-vision, values and commitments
Method, part 1: Visioning

Identify the core story linking everyone in the shared-enterprise

• What is everyone concerned about?
  • (TED example: Ideas)

• What is everyone doing about the concern?
  • (TED example: Spreading ideas)

• Why is this important to everyone?
  • (TED example: Ideas worth spreading)

Use this ‘vision-story’ as the core anchor for the quality-system
Method, part 2: Values-derivation

Derive values, laws, standards, metrics etc from the vision-story

• TED example “Ideas worth spreading”:
  • What is or is not “an idea that is worth spreading”? 
  • How would you measure the worth of an idea? 
  • What is or is not ‘good spreading’ of an idea? 
  • How would you measure the quality of spreading of ideas? 
  • What rules, laws, standards etc apply to spreading of ideas?

All of these help to indicate what ‘better’ would look like – and how to measure it, quantitatively or qualitatively
Method, part 3: Stakeholder-mapping

Verify vision, values etc with all of these stakeholder groups:

• Internal
• Transaction
• Direct-interaction *(market)*
• Indirect-interaction *(shared-enterprise)*
2. What structures to support ‘better’?

Requirement:
Structures that work the same way everywhere and that link to and support the chosen ‘better’
Structure: Service Canvas

values ('why')

value-flow ('how', 'with-what')

profit (money and more)
Structure: Validation-services

Value examples:
• Safety, security, efficiency, reliability, taste, robustness

For each value:
• Build awareness
• Build capability
• Enact at run-time
• Audit, review, improve
Structure: Service Cycle
3. What discipline is needed for change?

Requirement:
Methods that work the same way everywhere and that support the chosen ‘better’
The structure of a very incomplete task
The structure of a still-incomplete task
The structure of a complete task

- **Context**: check
- **Scope**: check
- **Plan**: sub-tasks, work-instructions with expected variance
- **Action**: work-instructions, actual variance, insights arising
- **Review**: outcomes and variances of sub-tasks, benefits realised, lessons learned, tasks arising

**Do the right things**

**Do things right**

**Learn**
Method: Change-mapping

- Step-by-step, consistent, everywhere
  - *(Mission-start)*
  - Context
  - Scope
  - Plan
  - Action
  - Review
  - *(Mission-end)*
- Fractal, linked, nested, same every instance
- Every tool is a plug-in, called in the same way
4. How do we verify and improve?

Requirement:
Methods to test and verify alignment to ‘better’ and give guidance for continual improvement
Method: After Action Review

- Apply after every action
- Benefits-realised, lessons-learned
- Test against ‘better’
- Derive change-tasks

- Two rules:
  - no blame
  - ‘pin your stripes at the door’
Why is this important?

A systematic, structured approach to ‘Change for the better’ supports continual improvement across the entire enterprise.

‘Change for the better’ is not a platitude, but a call to action

If we don’t do this, we face increased enterprise risks:
• Inconsistent alignment to enterprise values and goals
• Misalignment between business units and organisational silos
• Potential clashes with customers, suppliers and regulators
Resources

Change-mapping
Connecting business tools to manage change

Tools for Change-mapping
Expanding the Change-mapping tool-set

Advanced Change-mapping
Connecting business tools to manage change
Thank You

Tom Graves, Principal Consultant, Tetradian
Website: Tetradian.com
Email: info@tetradian.com