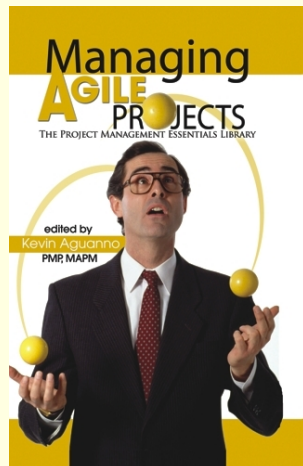
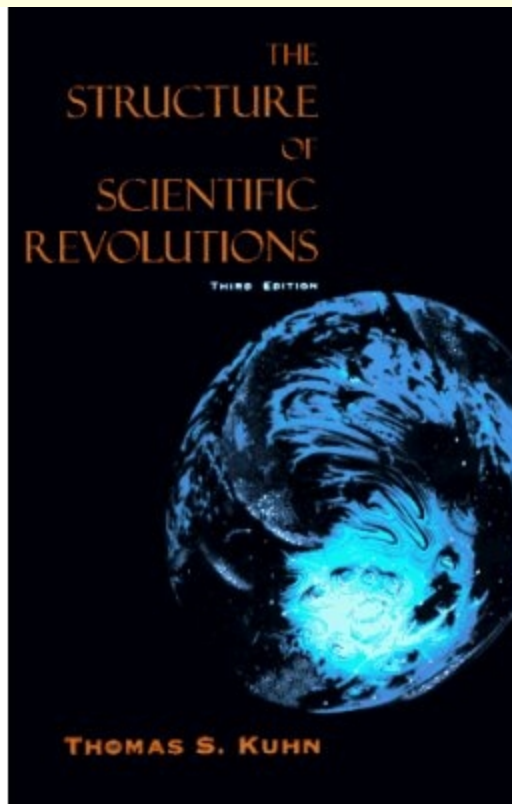


Stealth Methodology Adoption



Kevin Aguanno, PMP, MAPM
Editor of
Managing Agile Projects

New Approaches Usually Meet with Strong Resistance



By Thomas S. Kuhn
Written in 1962

Kuhn coined “Paradigm Shift” to describe the impact on thinking caused by new scientific approaches.

Traditionalists:

- Marginalize proponents
- Express disbelief
- May express anger
- Eventually resign themselves to the new thinking as it gains critical momentum

In Software Development, we Work at the Edge of Chaos

- Chrysler Comprehensive Compensation (C3)
 - Early 1990s, unify 3 payroll systems for 86,000 employees. Highly complex, no COTS package meets requirements. Many stakeholders within Chrysler.
 - Existing project a failure, floundering in all the complexity and making no progress.
- Singapore Project
 - Highly complicated lending system
 - 2 years, 3500 pages of use cases, object model with hundreds of classes and thousands of attributes (but no methods)
 - But NO Code

Traditional Methods Aren't Helping

Lock Down Requirements

- Stopping change means unresponsive and possibly not delivering value.

Tighten Control

- Traditional methods fight market demands for speed and innovation.

Fall Back to the Contract

- Sticking to original plans leads to customer dissatisfaction

We Needed New Methods to Deal with High Levels of Change

- 1970: Waterfall Model
- 1975: Iterative Enhancement
- 1988: Spiral Development Model
- Early 1990s: RAD, Crystal Methods, Lean, DSDM
- 1993: Capability Maturity Model
- 1996: Scrum
- 1998: Extreme Programming (XP)
- 2001: Agile Manifesto
- 2002: Agile Modeling

Proponents of the New Methods Faced the Resistance Noted by Kuhn

- The new methods were opposing entrenched bureaucracy (“corporate best practices”)
- Proponents were seen as “revolutionaries” and were marginalized
- People attacked the new methods as being “out of control”

As a Result, Proponents had to “Sell” the New Methods

Kuhn noted that direct “sales” techniques of persuasion directly confronts the traditionalists and leads to more conflict.

Question: How best can we convince our organizations and teams to adopt new methods?

Answer: Focus on the Benefits.

Example:

Agile Methods Reduce Risk

- Risk of building the wrong thing
 - “what I asked for, but not what I meant/wanted”
 - Being responsive to changing requirements
- Risk of poor quality
 - More time to correct defects
- Risk of getting stuck in “design churn”
 - Build what you know first

Example:

Agile Methods Improve Control

- Frequent delivery of working code means progress is objectively measurable
- More chances for sponsor/stakeholders to provide early feedback to redirect project priorities where necessary
- Misunderstandings are surfaced earlier
- Can end project early and still get measurable benefits

Example:

Agile Improves Communications

- Co-located teams
 - Whiteboarding
 - Ability to look over each other's shoulders at a moment's notice to share ideas
 - Full communications (incl. 70% nonverbal)
- Daily meetings
 - Everyone is kept abreast of issues and where they can help each other out
- Daily customer/sponsor involvement (XP Onsite customer, Scrum's daily meetings)
 - Quick and easy escalation
 - Snap decision-making
 - Instant feedback

Lessons Learned from the Trenches

Pointers from those who have been selling and delivering new methods

Identify Stakeholders and Understand Their Relative Power

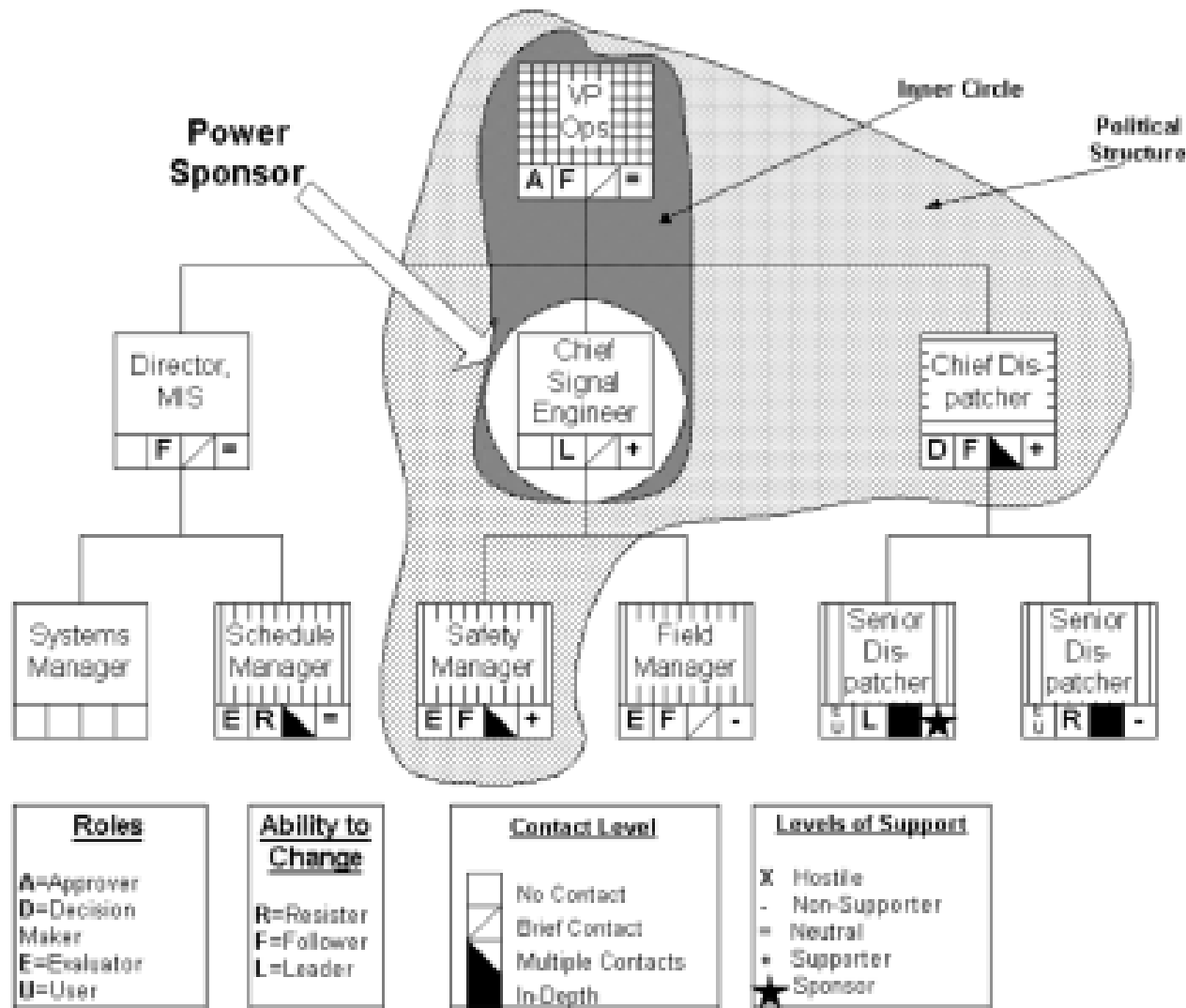


Diagram © 2003 Jeanne Barrecca

Understand Business Case for the Project

- Focus on business problems, not technology
- Speak to stakeholders in their own terms
- Look for ways new methods can support the business case through the benefits discussed earlier:
 - lower risk
 - higher quality
 - improved control
 - early delivery,
 - improved communications
 - etc.

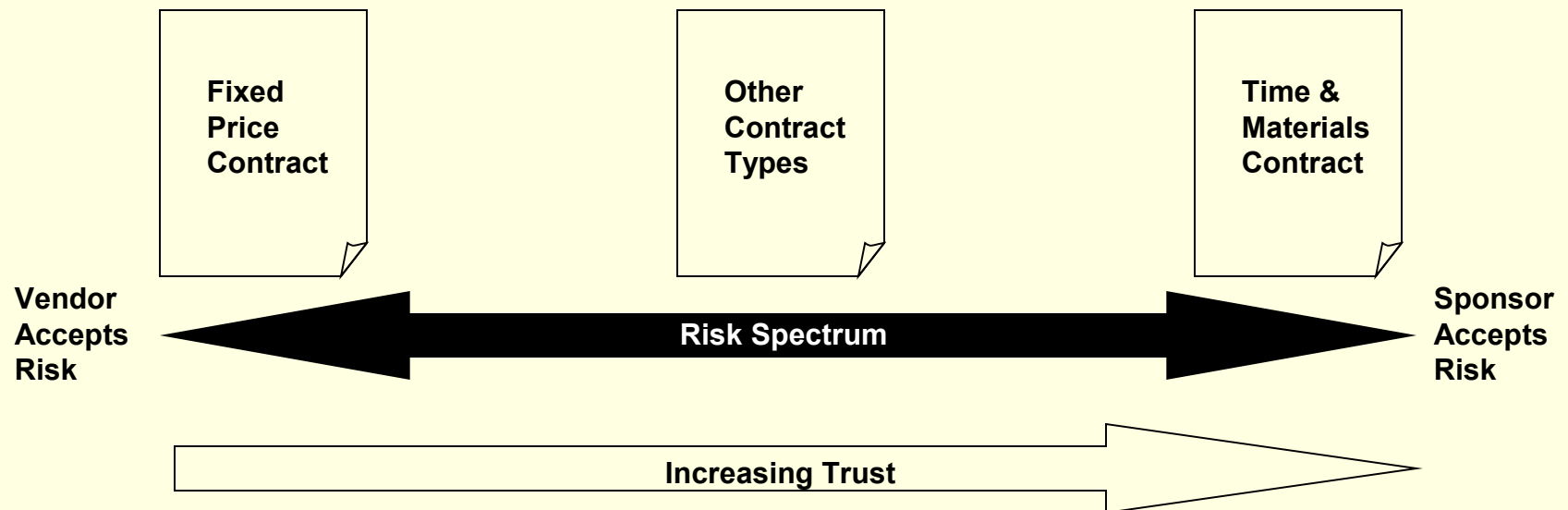
Use White Label Methodologies

- Don't talk to CFOs about “Extreme Programming”, the name will scare them off
- Don't get caught in methodology battles
- Instead, try to focus on individual techniques

Practice Stealth Methodology Adoption

- Don't adopt a whole new method at once; instead, adopt individual techniques into existing methods
 - Iterative Development (“iterations of waterfalls”)
 - Feature-Driven Development (PBS instead of WBS)
 - Test-Driven Development
 - Sponsor involvement in team planning and prioritization
 - Continuous Testing

Negotiate the Right Contracts



- Time & Materials (ideal for Agile projects)
- Fixed Price (fixed and variable scope, fixed number of complexity points, etc.)

Don't Start with Trying to Sell Contentious Items: Pair Programming

- Requires Trust

- Sponsors see it as doubling costs, wasting effort, etc.
- Developers may feel micromanaged or not trusted

- Instead, try this part time (or just in selected areas) and call it “early code inspections” or “continuous code inspections”

- Focus on improved quality (reduced errors) and shortened QA/fix time at the benefit.

Additional Resources

- Many resources available via the Agile Alliance Web Site:
www.agilealliance.com
- Kevin Aguanno (your speaker) is available for consultation at aguanno@ca.ibm.com. He is the author of several books related to this subject matter, including:

