

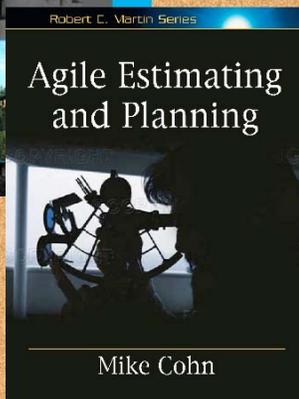
Succeeding With Agile: A Guide to Transitioning

Mike Cohn



1

Mike Cohn - background



- Agile coach and trainer**
- Founding member and director of Agile Alliance and Scrum Alliance
 - Founder of Mountain Goat Software
 - Ran my first Scrum project in 1995
 - Typical programmer to manager etc. progression



© 2003–2008 Mountain Goat Software®

2

Topics today...

1. Why transitioning to agile is hard
2. ADAPTING to agile development
3. A framework for transitioning
4. The role of leaders
5. Patterns of agile adoption



© 2003–2008 Mountain Goat Software®

3

Why Transitioning
to Agile
Is Hard



© 2003–2008 Mountain Goat Software®

4

Wednesday, May 7, 2008

1

Change is not top-down or bottom-up; it's both

- Two simplistic views of change:
 - Top down
 - Powerful leader shares a vision
 - Bottom-up
 - A team starts and everyone else sees the benefits of the new approach
- But, transitioning to agile is neither top-down nor bottom-up
 - It's everywhere, all together, all-at-once



© 2003–2008 Mountain Goat Software®

5

2

Best practices are tempting

- It is tempting to codify things that work in a given context into *best practices*
 - This leads to inflexible processes[†]
- Once we know what's "best" we stop adapting
 - Or even thinking about what we're doing
- Once we've stopped inspecting and adapting we're not agile, or won't be for long



[†]Source: Anderson, P. "Seven Layers for Guiding the Evolving Enterprise" in *The Biology of Business*.

© 2003–2008 Mountain Goat Software®

6

3

The transition process must be congruent with the development process

Part of the move to agile is a move to self-organizing teams

Moving to self-organization requires self-organization



© 2003–2008 Mountain Goat Software®

7

4

We cannot predict how an organization will respond to change

- How we traditionally view our organizations:
 - Behavior is highly predictable
 - Once set in motion, will continue in motion
- An organization change strategy can be mapped out:
 - Do this first, then that, then such and so
 - And we'll end up right where I predict



© 2003–2008 Mountain Goat Software®

8

“From a very early age, we are taught to break apart problems, to fragment the world. This apparently makes complex tasks and subjects more manageable, but we pay a hidden, enormous price. We can no longer see the consequences of our actions; we lose our intrinsic sense of connection to a larger whole. When we try to ‘see the big picture,’ we try to reassemble the fragments in our minds, to list and organize all the pieces. But, as physicist David Bohm says, the task is futile—similar to trying to reassemble the pieces of a broken mirror to see a true reflection. Thus, after awhile we give up trying to see the whole altogether.”

Peter Senge, *The Fifth Discipline*



© 2003–2008 Mountain Goat Software®

9

“This machine imagery [Newtonian view] leads to the belief that studying the parts is the key to understanding the whole. Things are taken apart, dissected literally or figuratively...and then put back together without any significant loss. The assumption is that the more we know about the workings of each piece, the more we will learn about the whole.”

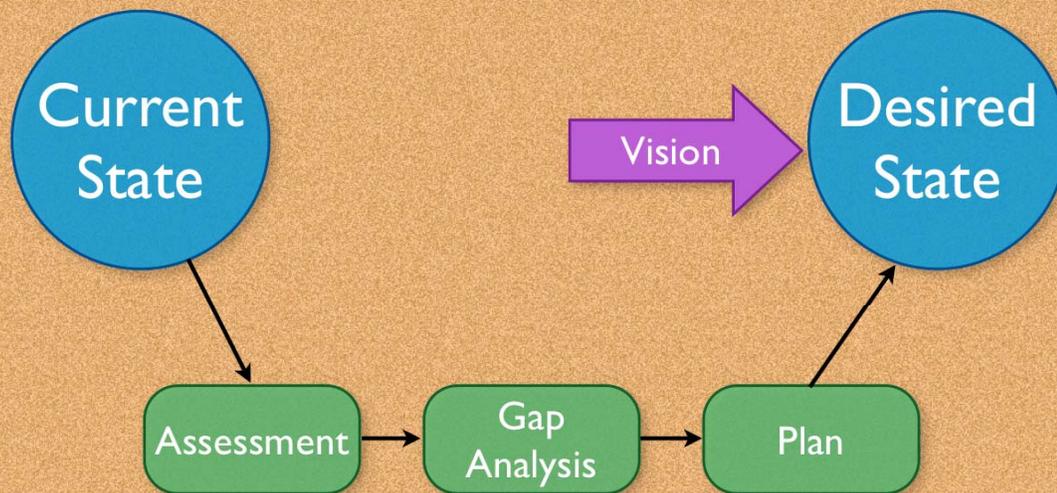
~Margaret Wheatley
in *Leadership and the New Science*



© 2003–2008 Mountain Goat Software®

10

The Newtonian view leads to thinking of change like this



© 2003–2008 Mountain Goat Software®

11

We need a different mental model

- The organization as a **C**omplex **A**daptive **S**ystem (CAS)

- A dynamic network of many agents
 - acting in parallel
 - acting and reacting to what other agents are doing
- Control is highly dispersed and decentralized
- Overall system behavior is the result of a huge number of decisions made constantly by many agents

John Holland in *Complexity: The Emerging Science at the Edge of Order and Chaos* by Mitchell Waldrop



© 2003–2008 Mountain Goat Software®

12

Differing views of success

Newtonian view

Success =
closing the gap with the
desired state

CAS view

Success =
achieving a good fit with
the environment

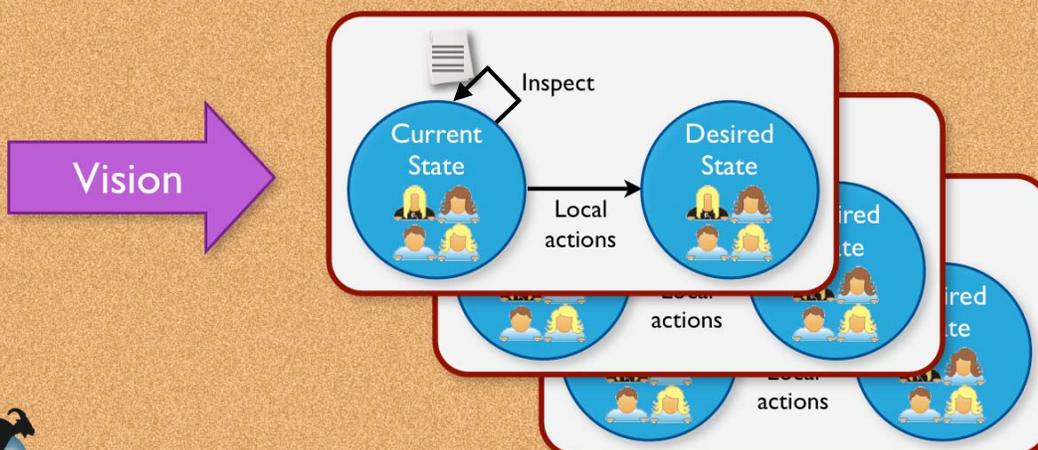


© 2003–2008 Mountain Goat Software®

13

Local goals and gaps

- Local agents (individuals, project teams, discipline coworkers) identify local gaps based on their local goals



© 2003–2008 Mountain Goat Software®

14

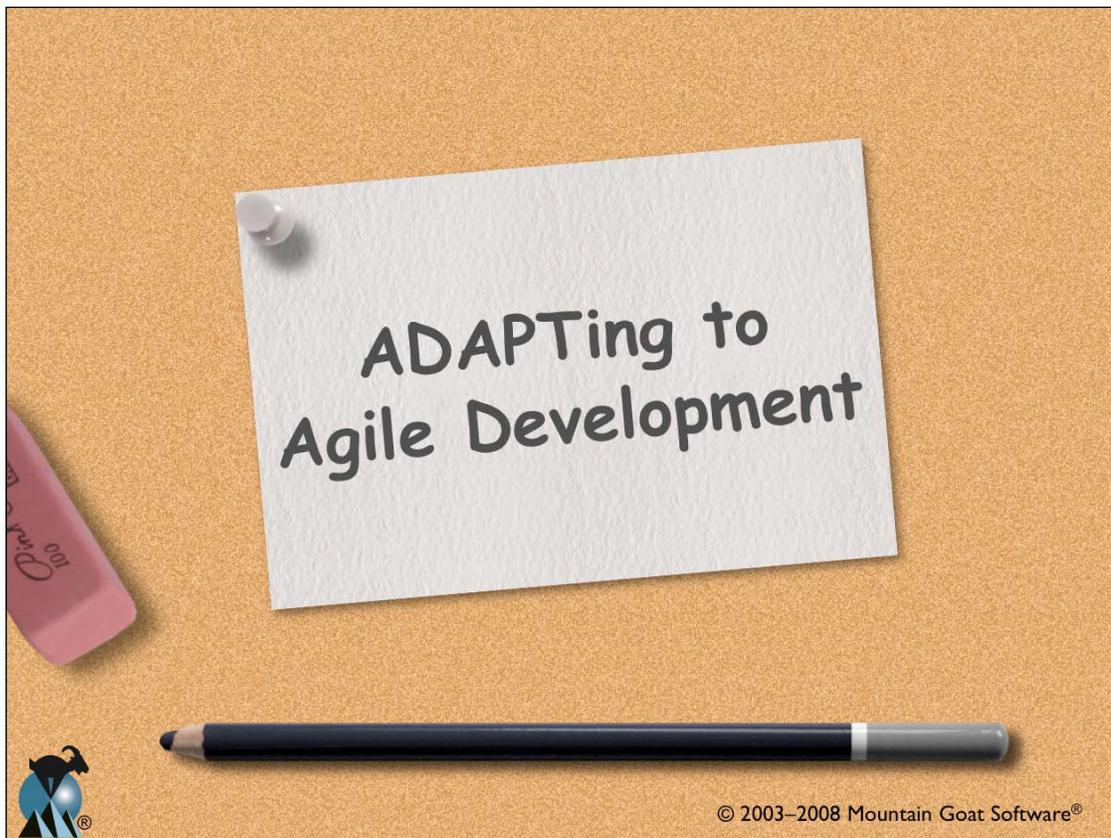
Traditional model of change	Complex, adaptive model of change
Behavior is predictable and controllable	Behavior is unpredictable and uncontrollable
Direction is determined by a few leaders.	Direction is determined through emergence and by many people
Every effect has a cause	Every effect is also a cause
Relationships are directive	Relationships are empowering
Efficiency and reliability are measures of value	Responsiveness to the environment is the measure of value
Decisions are based on facts and data.	Decisions are based on patterns and tensions.
Leaders are experts and authorities.	Leaders are facilitators and supporters.



Adapted from Olson and Eoyang, *Facilitating Organization Change*.

© 2003–2008 Mountain Goat Software®

15



© 2003–2008 Mountain Goat Software®

16

Wednesday, May 7, 2008

A

Awareness that the current approach isn't working

D

Desire to change

A

Ability to work in an agile manner

P

Promote early successes to build momentum and get others to follow

T

Transfer the impact of agile throughout the organization so that it sticks



© 2003–2008 Mountain Goat Software®

17

I'm the ScrumMaster and...

...the developers are not meeting expectations for code quality. One of our challenges is that we're still hacking our way through lots of legacy code that isn't unit-testable or automated yet, but is mission critical and the person who has been working mostly on that area of code consistently leaves holes in the design and implementation of new pieces of that code. We also have the issue with at least one other developer as well.

1. Is this a problem of Awareness, Desire or Ability?
2. Thinking about ADAPT, what might you try?



© 2003–2008 Mountain Goat Software®

18

Tools for building...

Awareness

- Communicate
- Establish a vision
- Narrow the focus
- Metrics
- Run a pilot

Desire

- A clear vision (again)
- Share examples of success
- Public praise for the right behavior
- Align incentives
- Turn the transition over to individuals
- Build momentum



© 2003–2008 Mountain Goat Software®

19

Tools for building...

Ability

- Pairing (of all sorts)
- Bring in outside coaches
- Develop your own internal coaches
- Formal training
- Practice



© 2003–2008 Mountain Goat Software®

20

Promote

- Celebrate and share even small, early wins
- Goal is to build momentum
 - Want a feeling of inevitability around the transition
- Reduce upcoming resistance before it starts
- Send people on an agile safari
- Attract attention and interest



© 2003–2008 Mountain Goat Software®

21

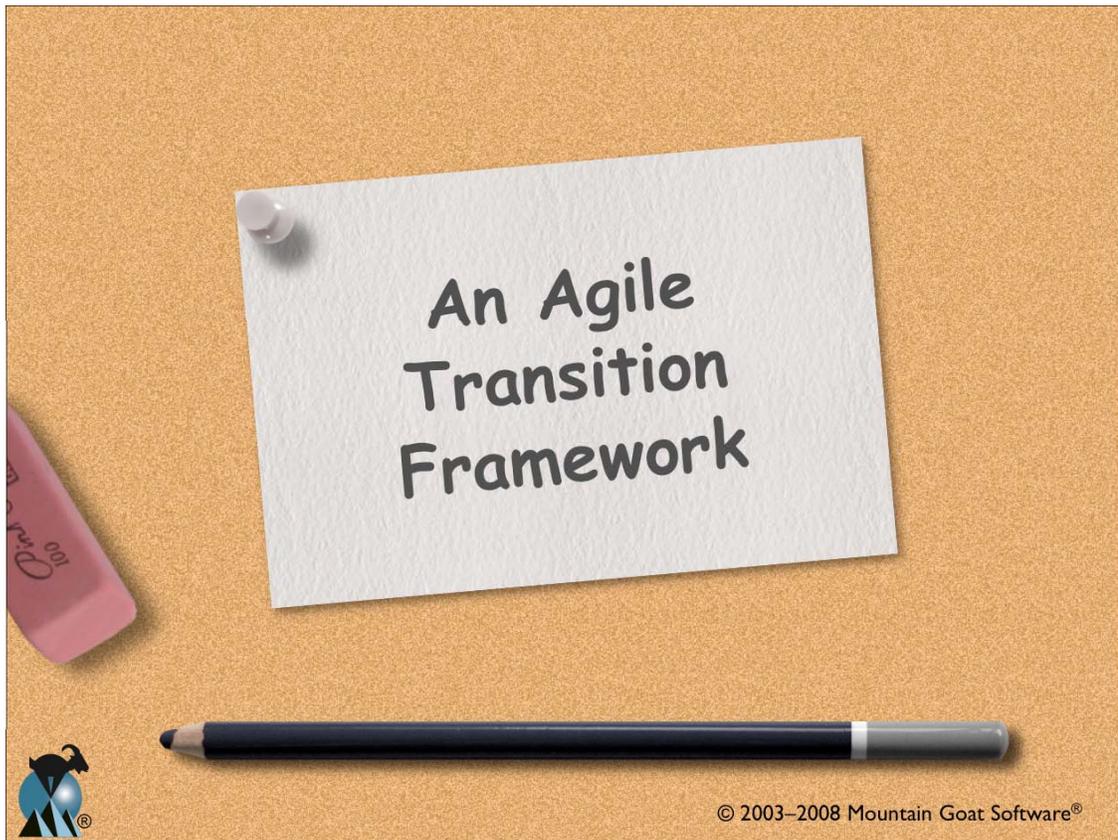
Transfer

- Transfer the effects of agile beyond the current group
 - A team transfers to its department
 - A department transfers to its division
 - etc.
- If you don't transfer, the transition will eventually and inevitably fail
 - Too much **organizational gravity** pulling us back toward the status quo
- Example:
 - If you don't align promotions, raises, annual reviews, those will work against you



© 2003–2008 Mountain Goat Software®

22



23

- On projects we learn we cannot precisely anticipate:
 - our users' requirements
 - how long it will take to develop a feature or entire system
 - which design will be best
 - the set of tasks necessary to develop a feature
- So we devise alternative approaches:
 - Rather than ask for upfront specs, we deliver partial solutions, solicit feedback, and repeat
 - Rather than design the whole system, we design incrementally and adjust based on what we learn

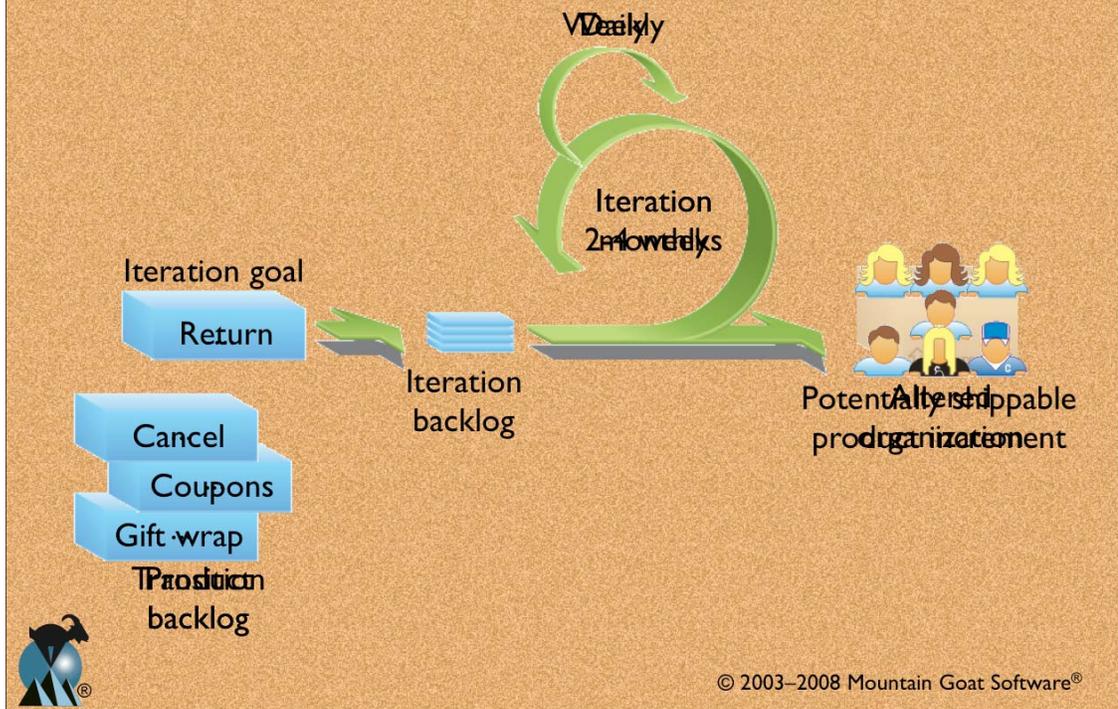
We need to do the same for the transition effort



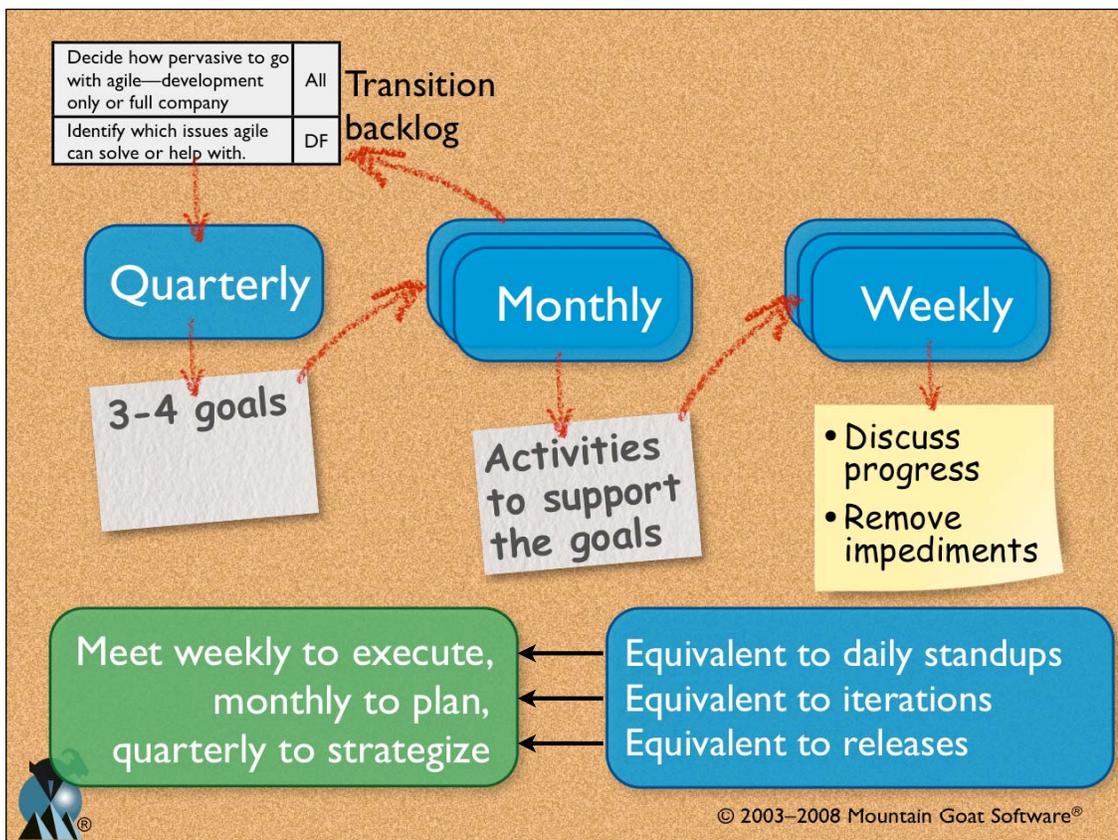
© 2003–2008 Mountain Goat Software®

24

An agile transition process



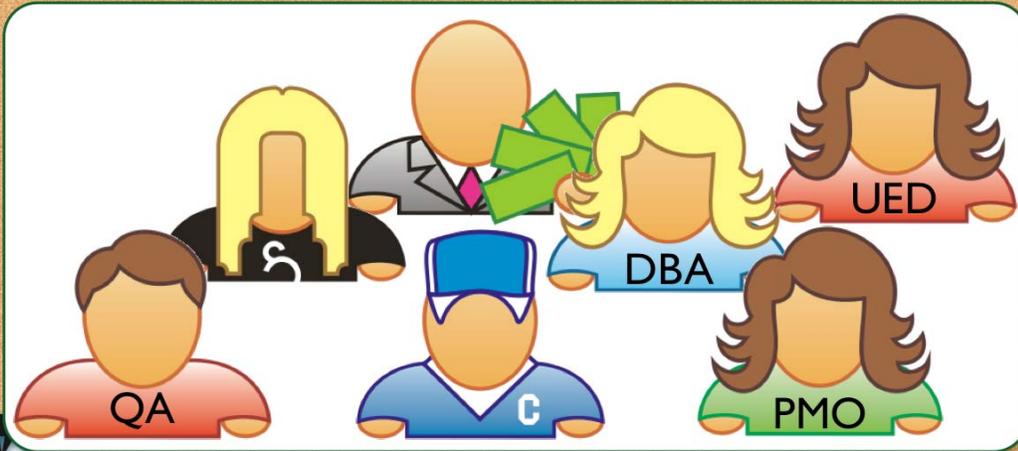
25



26

Establish a “guiding coalition”

- Who?
 - Sponsor—senior person responsible for success
 - Area managers or leads who can make it happen



© 2003–2008 Mountain Goat Software®

27

Who should *not* be on the guiding coalition

- People with big egos
 - Big egos fill the room; leave little space for others
 - Don't understand their own limitations
- Snakes
 - Someone who poisons relationships among team members
- Reluctant participants
 - Lack time or enthusiasm
 - But may have needed expertise or political clout



© 2003–2008 Mountain Goat Software®

28

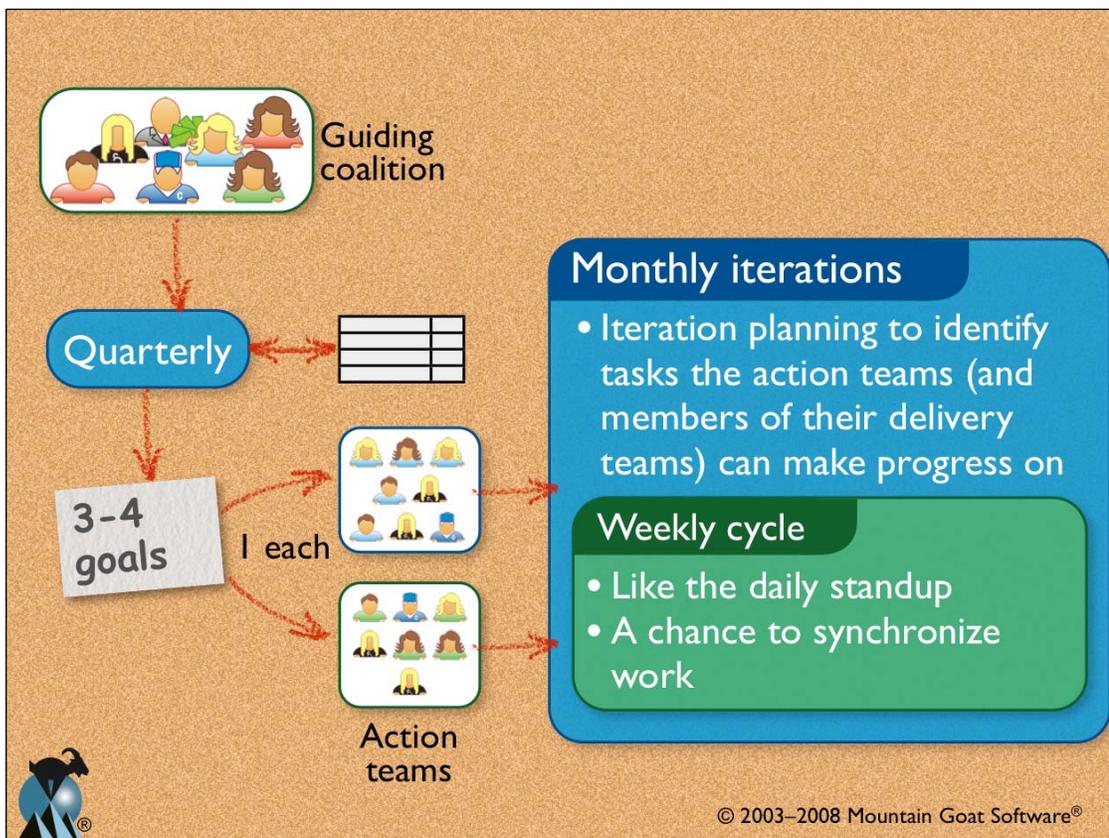
Action teams

- Usually more than one at any time
 - Each focused on a different goal
- Organized around achieving specific organizational goals
 - e.g., test automation or user experience design
- Some teams in an organization will be organic
 - Individuals notice something needs to be achieved
- Others will be formally-sponsored
 - Guiding coalition puts someone in charge of achieving a goal that hasn't been picked up
 - Usually best only if an organic team doesn't form



© 2003–2008 Mountain Goat Software®

29



© 2003–2008 Mountain Goat Software®

30

Action team members

- Try to form these teams organically
 - Possible with a point person to start the team
 - True product owner for the team is the guiding coalition
 - But this starting person acts as a combination day-to-day product owner and ScrumMaster
- Initial membership
 - Start with 1-3 members who “get it”
 - Ask each of those members to pick 1-2 more



© 2003–2008 Mountain Goat Software®

31

Action team member considerations

- Think about
 - Who has the power to make or break the transition to agile?
 - Who controls critical resources or expertise?
 - How will each be affected?
 - How will each react?



© 2003–2008 Mountain Goat Software®

32

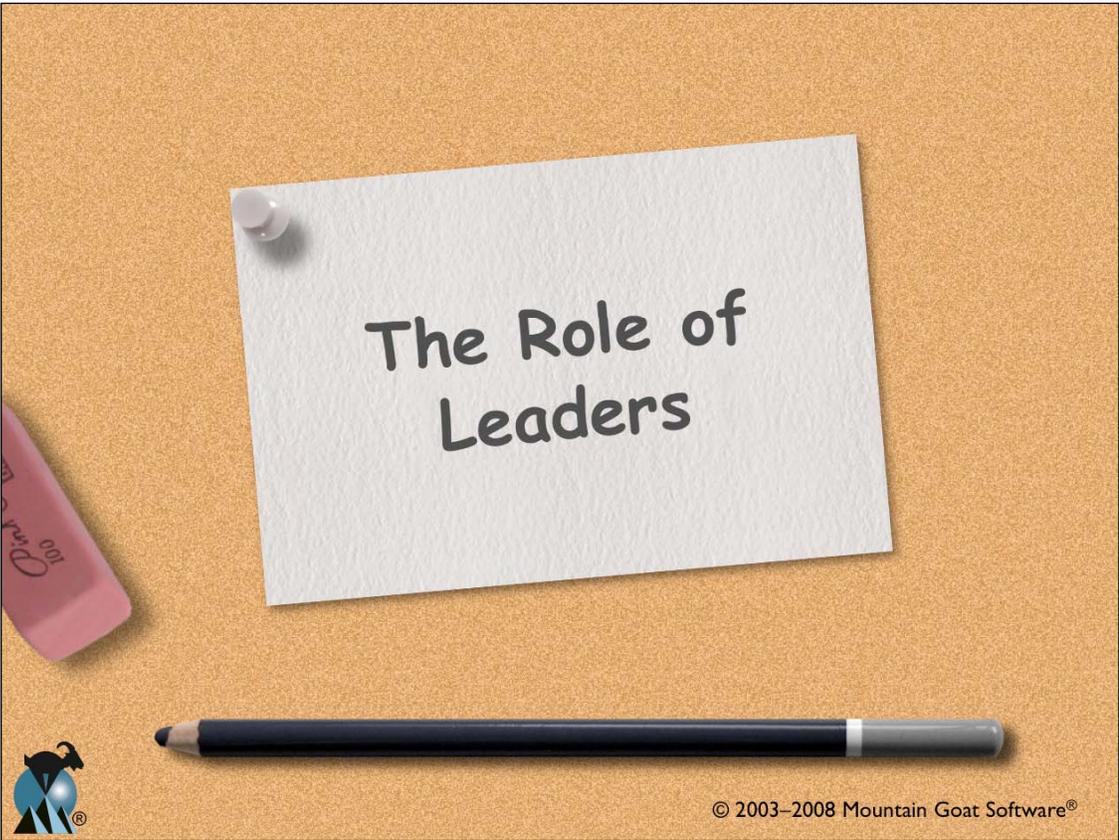
Additional considerations

- Who will gain or lose something by the transition to agile?
- Are there blocs likely to mobilize against or in support of the transition?
- Do team members have sufficient credibility that the teams' opinions and results are taken seriously?
- Can team members put their personal interests aside in favor of the organizational goal?



© 2003–2008 Mountain Goat Software®

33

A corkboard with a white sticky note pinned to it. The sticky note has the text "The Role of Leaders" written on it. A black pencil and a pink eraser are also visible on the board.

The Role of
Leaders



© 2003–2008 Mountain Goat Software®

34

Leading an agile transition

- Action team and other formal leaders must lead the transition
 - but cannot do so in the usual ways
- Self-organizing groups still require leadership
- Lead through example, questions, and focus
 - “Nudge” the organization; Poke and prod;
 - See how the organization responds



© 2003–2008 Mountain Goat Software®

35

Pre-requisites of self-organization

Container

- A boundary within which self-organization occurs
 - Company, project, team, city, role, nationality

Differences

- There must be differences among the agents acting in our system
 - Technical knowledge, domain knowledge, education, experience, power, gender

Transforming Exchanges

- Agents in the system interact and exchange resources
 - Information, money, energy (vision)



Glenda Eoyang: *Conditions for Self-Organizing in Human Systems*

© 2003–2008 Mountain Goat Software®

36

Using the CDE model

- When stuck thinking about how to nudge the organization think of the:
 - **C**ontainers
 - formal teams, informal teams, clarify (or not) expectations
 - **D**ifferences
 - Dampen or amplify them within or between containers
 - **E**xchanges
 - Insert new exchanges, new people, new techniques or tools



© 2003–2008 Mountain Goat Software®

37



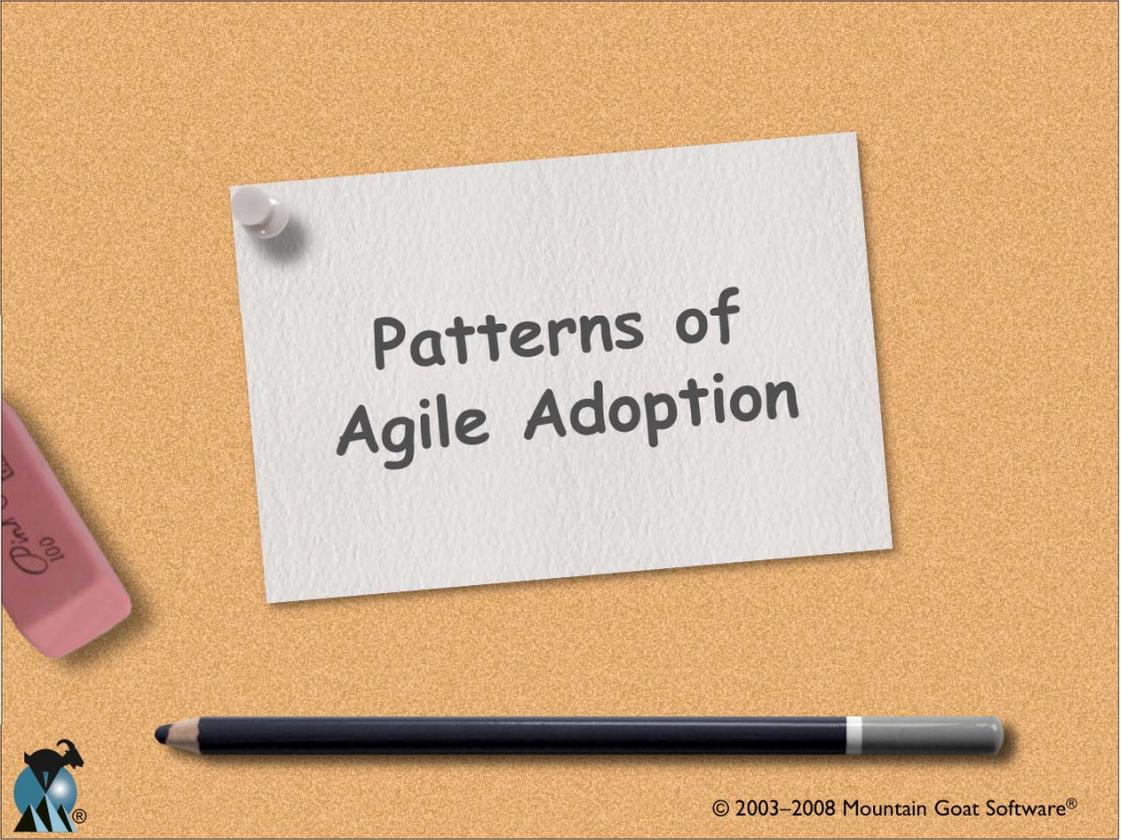
The team consists of four developers, two testers, a database engineer and you. The developers and testers are not working well together. Developers work in isolation until two days are left in the iteration. They then throw the code “over the wall” to the testers.

Would you change a Container, Difference or Exchange to address this issue?



© 2003–2008 Mountain Goat Software®

38



Patterns of Agile Adoption

© 2003–2008 Mountain Goat Software®

39

Six patterns, three decisions

Technical Practices
First

or

Iterative First

Start Small

or

All In

Stealth Mode

or

Public Display of
Agility



© 2003–2008 Mountain Goat Software®

40

Technical Practices First

Advantages

- Very rapid improvements are possible
- The transition can be quick

Disadvantages

- Technical practices support each other in subtle ways
- There is likely to be strong resistance to some practices
- Outside coaching will likely be needed

Useful when

- The most pressing issues facing the project are ones that can be solved with technical practices.
- You aren't starting a huge number of teams at once
- Team members have solid technical backgrounds
- There is a desperate need to improve



© 2003–2008 Mountain Goat Software®

41

Iterative First

Advantages

- It's easy to start
- It's hard to argue against

Disadvantages

- The team may not choose to add the technical practices

Useful when

- You want to transition more than a handful of teams concurrently
- You are starting with a stalled project
- Lots of different technologies are in use by various teams



© 2003–2008 Mountain Goat Software®

42

Start Small

Advantages

- Cost of mistakes is minimized
- You can almost guarantee success

Disadvantages

- Conclusions may not be compelling
- It takes a lot of time
- Agile teams will need to work with non-agile teams

Useful when

- There is reluctance to commit fully to agile
- The risks of failing an all-at-once transition outweigh the advantages
- You can afford the time it takes



© 2003–2008 Mountain Goat Software®

43

All In

Advantages

- It's over quickly
- There's no organizational dissonance from using two processes at once
- It can reduce some resistance

Disadvantages

- It's risky
- It's costly
- It will likely require a reorganization

Useful when

- You want to send a clear message
- Time is critical
- Your team isn't too small or too big



© 2003–2008 Mountain Goat Software®

44

Stealth Mode

Advantages

- There's no additional pressure
- No one knows about it until you tell them
- No one can tell you not to do it

Disadvantages

- You won't have any organizational support
- Skeptics will only hear about success, they won't witness it

Useful when

- You want to experiment
- You don't have any organizational support
- You expect strong resistance



© 2003–2008 Mountain Goat Software®

45

Public Display of Agility (PDA)

Advantages

- Everyone knows you're doing it so you're more likely to stick with it
- It establishes a vision to work toward
- Makes a firm statement that you are committed to transitioning

Disadvantages

- Announcing something before you do it can make you look foolish
- Resisters will come out of the woodwork

Useful when

- You are confident in the approach and committed to achieving it
- You are likely to face stiff resistance and want to face it all at once



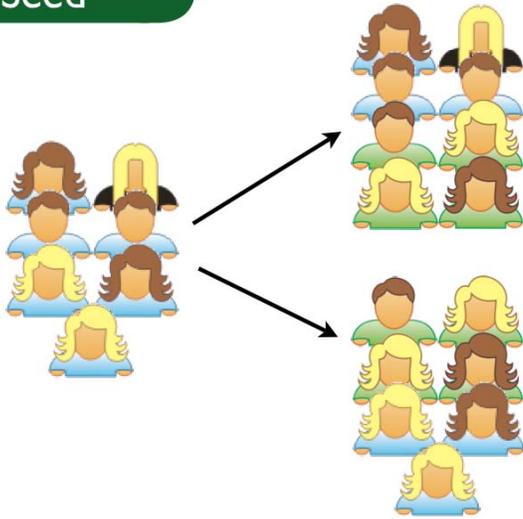
© 2003–2008 Mountain Goat Software®

46

Three expansion patterns

Split and Seed

1

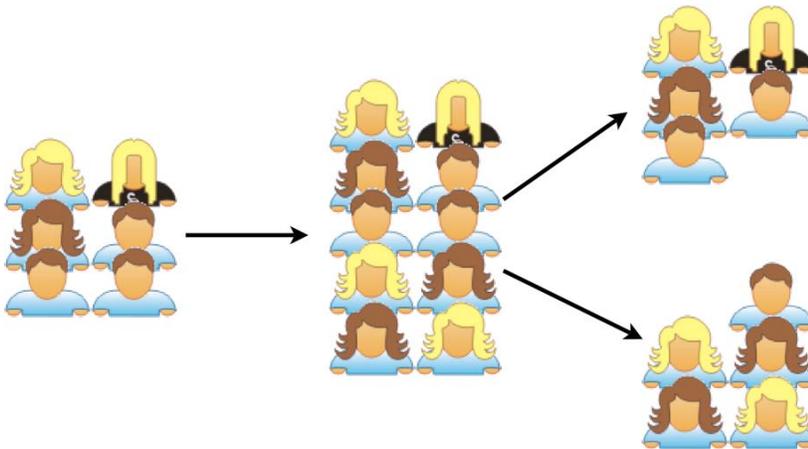


© 2003–2008 Mountain Goat Software®

47

Grow and split

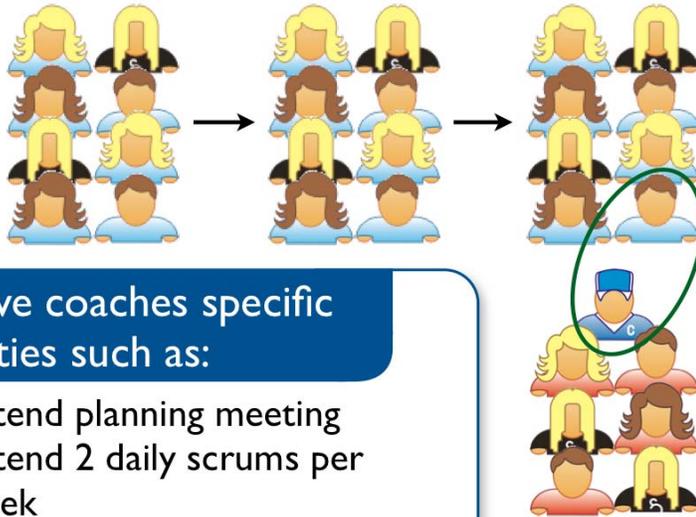
2



© 2003–2008 Mountain Goat Software®

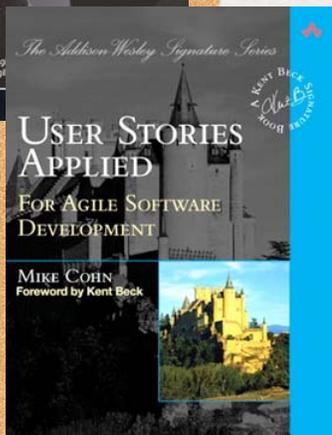
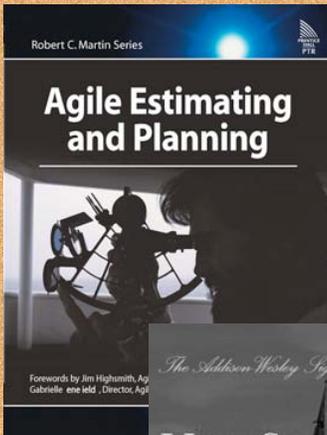
48

Internal coaching



Give coaches specific duties such as:

- Attend planning meeting
- Attend 2 daily scrums per week
- Spend 4 hours with the team per sprint



Mike Cohn
 mike@mountaingoatsoftware.com
 www.mountaingoatsoftware.com
 (720) 890-6110 (office)
 (303) 810-2190 (mobile)

