Agile Logic
Deliver Better Software Faster

Agile Logic
Deliver Better Software Faster

The Essence of Agile

A Tour of Agile Methods

Agile / XP So. Cal. Meeting – 1/21/09

Speaker: Paul Hodgetts, Agile Logic, www.AgileLogic.com

Rev 080106

Your Speaker: Paul Hodgetts

- Team coach, trainer, consultant, developer
- Founder and CEO of Agile Logic (based in Fullerton)
- 25 years overall, 10 years agile experience
- Certified Scrum Trainer
- Focus on Enterprise use of agile
- Author (Extreme Programming Perspectives)
- Presenter at conferences (Agile 200x, SD West, JavaOne)
- Agile Alliance Program Director
- Member of CSUF agile advisory board
- Contact info: phodgetts@agilelogic.com

Agile Logic
Deliver Better Software Faster

Agile Logic
Deliver Better Software Faster

Agile Software Development Approach

Agile Security Infrastructure

Agile

Agile

Agile

Agile, adj. 1. quick and well-coordinated in movement; nimble. 2. active; lively. 3. marked by an ability to think quickly

Agile

Agile SOA Architecture

Agile

The Agile Enterprise

Agile

Agile

Agile

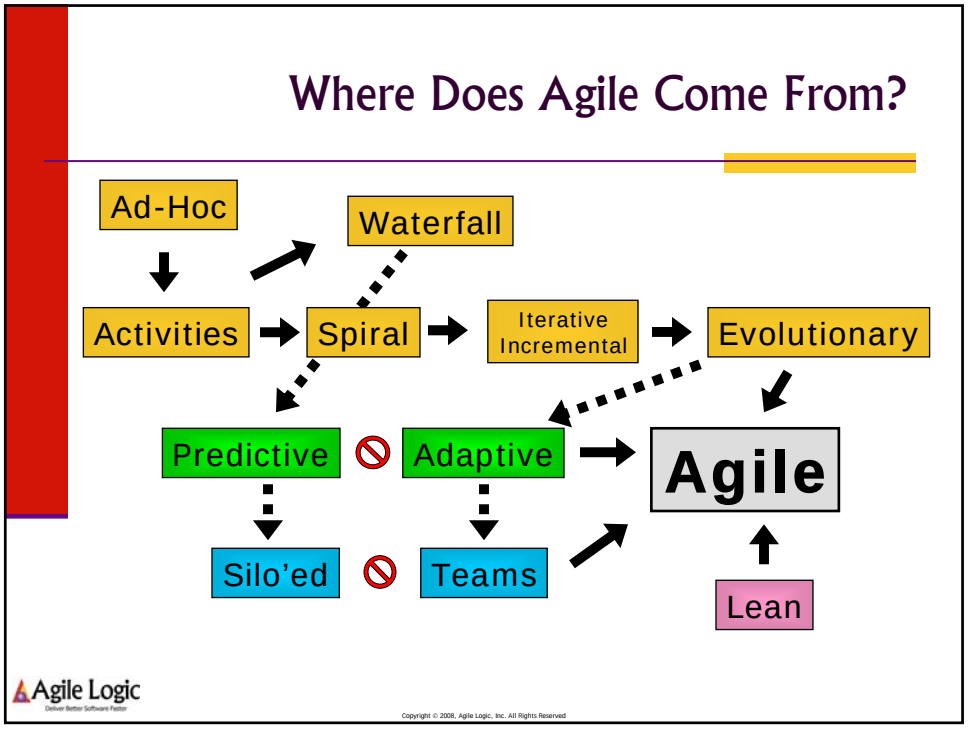
Agile Requirements Traceability Tool

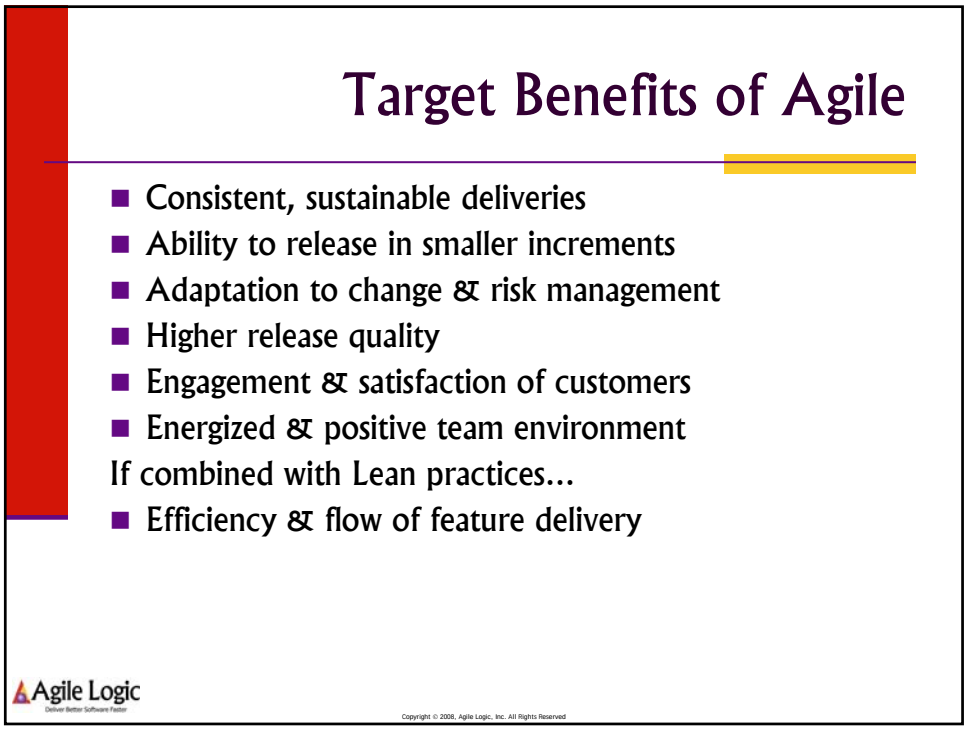
Agile

Agile Logic

What Makes Up our "Approach"?

Agile Logic



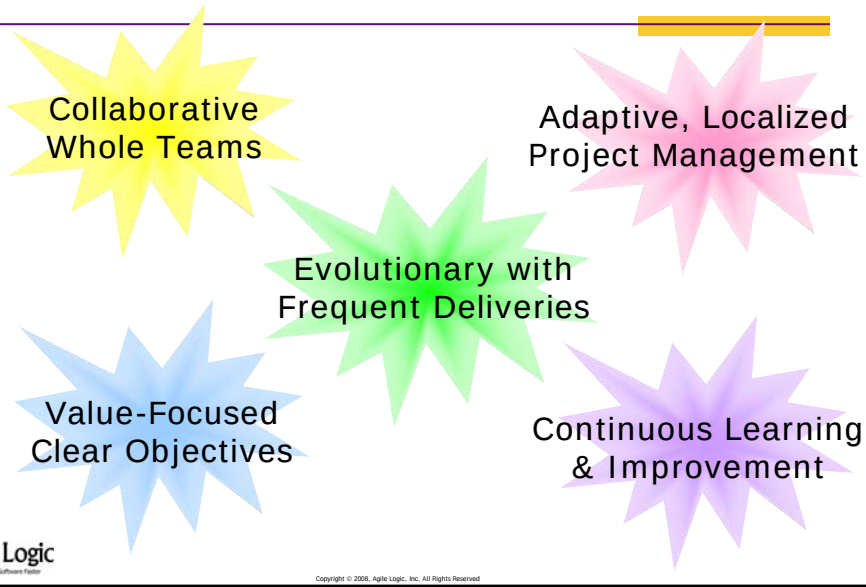


Underlying Agile Values

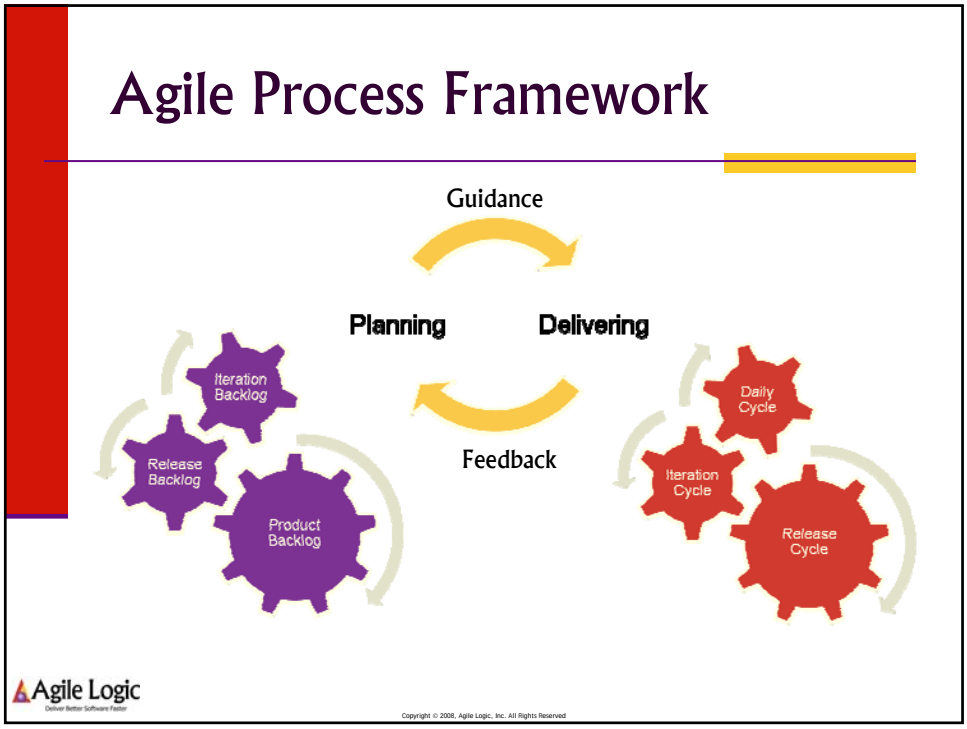
- Delivering stakeholder value is what matters
- Trust people working together
- Change will happen, we must adapt
- Frequent, concrete feedback is the truth
- Delaying risk is… too risky

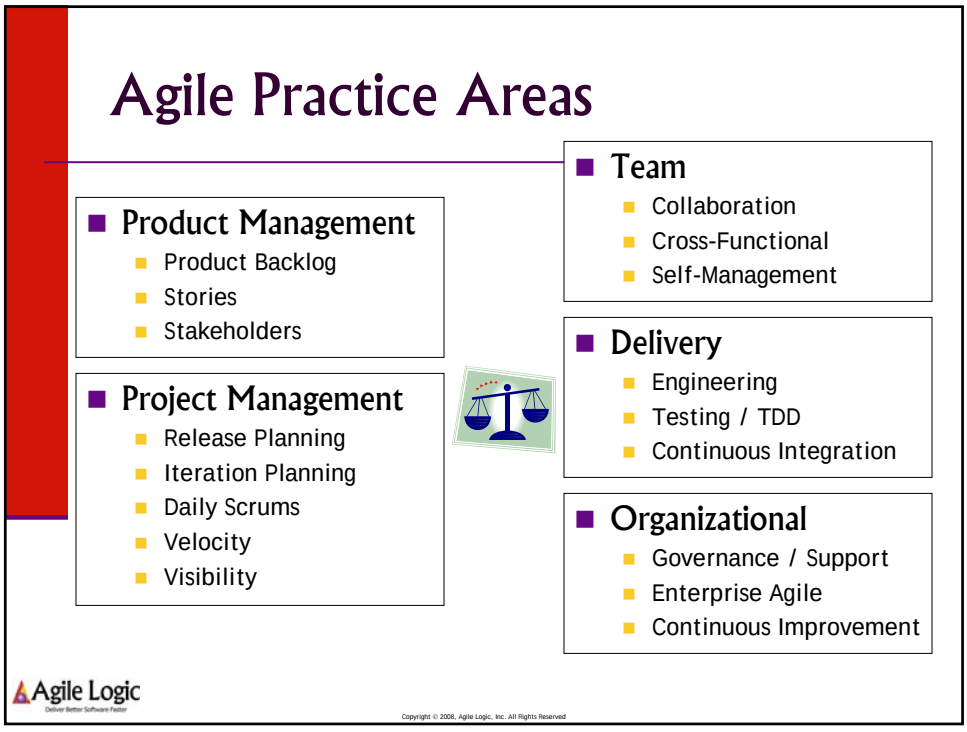
Agile Logic

Key Agile Strategies

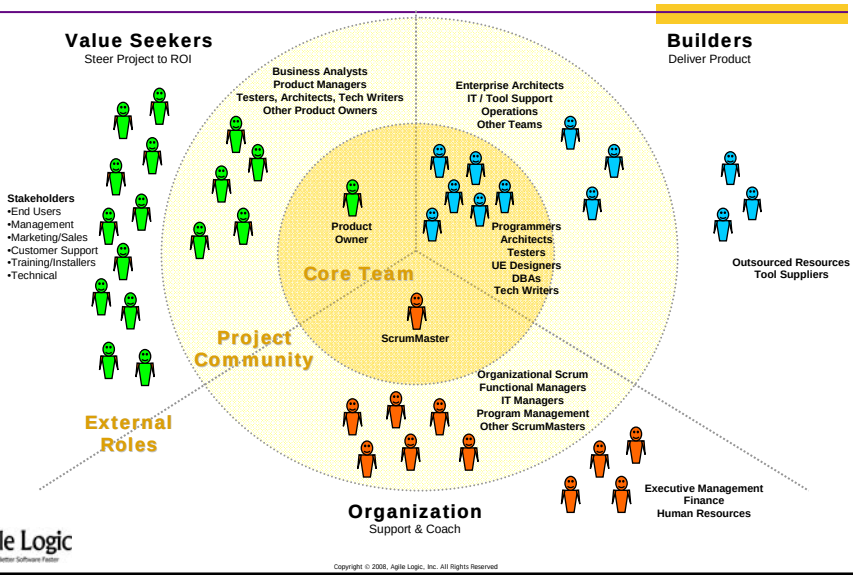


Agile Logic





Agile Team Structure



Agile Methods

- Provide a collection of:
  - Values
  - Strategies / Structure
  - Practices / Policies

- Reflect the experiences of the founder(s)
- None are "complete" processes
  - Processes are context-specific
  - Each agile method is a starting point
  - Each method adds to the "tool kit"

Scrum

- Originated by Jeff Sutherland 1994
- Ken Schwaber formalized in 1996
- Scrum Alliance 2003
- Named after ???

Agile Logic

Scrum Approach

- Empirical processes, "inspect and adapt"
- Empowered teams
- Provides a framework to create visibility, focus
  - Not prescriptive about disciplinary practices
- Requires removing impediments
- Scrum values:
  - Commitment
  - Focus
  - Openness
  - Respect
  - Courage

Agile Logic

Scrum Practices

- Product Backlog
- (Strategic/Release Planning)
- Sprints (30 days or less)
  - Sprint Planning -> Sprint Backlog
  - Daily Scrum
  - Sprint Review
  - Sprint Retrospective
- Each Sprint must deliver "done" product

Agile Logic

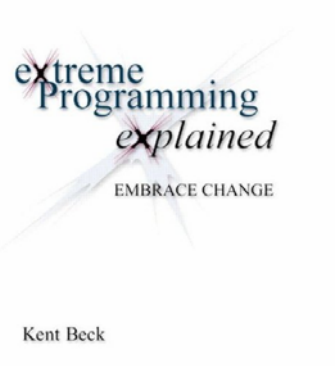
Scrum Roles

- Scrum Team (~7 team members)
  - Must contain all the needed resources
- Two identified roles in team:
  - ScrumMaster
    - Process implementation and improvement
  - Product Owner
    - Product Backlog and maximizing value
    - Represents Stakeholders

Agile Logic

Extreme Programming (XP) 1st ed.

- Kent Beck, Ward Cunningham 1980s
- Chrysler C3 project mid-1990s, Ron Jeffries
- Published 1999
- Named after ???
- Large internet community



Agile Logic

XP Approach

- Emphasizes:
  - Rapid creation of high-value software
  - Skillful and sustainable techniques
  - Low formality and low ceremony
  - Responsiveness to change
- XP Values:
  - Communication
  - Simplicity
  - Feedback
  - Courage

Agile Logic

Agile Logic
Deliver Better Software Faster

XP Practices

- Planning Game
- Small Releases
- System Metaphor
- Simple Design
- Testing
- Refactoring
- Pair Programming
- Collective Ownership
- Continuous Integration
- 40-Hour Week
- On-Site Customer
- Coding Standards

Agile Logic
Deliver Better Software Faster

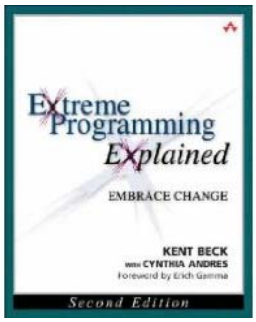
XP Roles

- Separation of technical vs. business decisions
- XP roles:
  - Programmer
  - Customer
  - (Tester)
  - (Tracker)
  - (Coach)
  - (Consultant)
  - (Big Boss)

Agile Logic
Deliver Better Software Faster

Extreme Programming (XP) 2nd ed.

- Kent Beck's follow-on 2004
- Did not generate as much interest
- Extensions widely practiced



Agile Logic

Changes to Approach

- Not as prescriptive as 1st ed.
- Expanded practices
  - Recognizes tailoring the process
- Additional value:
  - Respect

Agile Logic

Agile Logic
Deliver Better Software Faster

Primary XP Practices

- Sit Together
- Whole Team
- Informative Workspace
- Energized Work
- Pair Programming
- Stories
- Weekly Cycle
- Quarterly Cycle
- Slack
- Ten-Minute Build
- Continuous Integration
- Test-First Programming
- Incremental Design

Agile Logic
Deliver Better Software Faster

Corollary XP Practices

- Real Customer Involvement
- Incremental Deployment
- Team Continuity
- Shrinking Teams
- Root-Cause Analysis
- Shared Code
- Code and Tests
- Single Code Base
- Daily Deployment
- Negotiated Scope
- Pay-Per-Use

Agile Logic
Deliver Better Software Faster

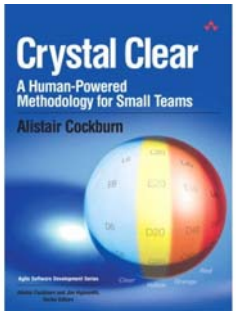
Expanded XP Roles (Whole Team)

- Testers
- Interaction Designers
- Architects
- Project Managers
- Product Managers
- Executives
- Technical Writers
- Users
- Programmers
- Human Resources

Agile Logic

Crystal (Clear)

- Alistair Cockburn
- From the study of projects while at IBM
- "Surviving Object-Oriented Projects" 1998
- "Agile Software Development" 2002
- "Crystal Clear" 2004



Agile Logic

Crystal Approach

- Core strategies:
  - Colocation of the team
  - Frequent delivery
  - Access to expert user
- Method tailored for size and criticality
  - Clear, yellow, orange, red

Crystal Properties

- Frequent Delivery
- Reflective Improvement
- Osmotic Communication
- Personal Safety
- Focus
- Easy Access to Expert Users
- Sound Technical Environment
  - Automated testing, config mgmt, integration

Crystal Strategies

- Exploratory 360°
- Early Victory
- Walking Skeleton
- Incremental Re-architecture
- Information Radiators

Agile Logic

Crystal Techniques

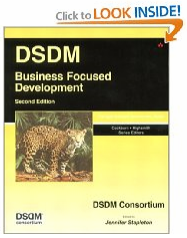
- Methodology Shaping
- Reflection Workshop
- Blitz Planning
- Delphi Estimation
- Daily Stand-Up Meetings
- Essential Interaction Design
- Process Miniature
- Side-By-Side Programming
- Burn Charts

Agile Logic

Agile Logic
Deliver Better Software Faster

DSDM / Atern

- RAD practices from early 1990s
- Originated in the UK, more popular in Europe
- Name:
  - Dynamic Systems Development Method
  - Dynamic Solutions Delivery Model
- DSDM Consortium
- Atern published 2008 (dsdm.org)



Agile Logic
Deliver Better Software Faster

DSDM Atern Principles

- Focus on the business need
- Deliver on time
- Collaborate
- Never compromise quality
- Build incrementally from firm foundations
- Develop iteratively
- Communicate continuously and clearly
- Demonstrate control

Agile Logic
Deliver Better Software Faster

Feature-Driven Development (FDD)

- Jeff De Luca, Peter Coad 1997-1998
- Generated from "Singapore Project"
- "Practical Guide" 2002
- Remains fairly obscure

FDD Practices

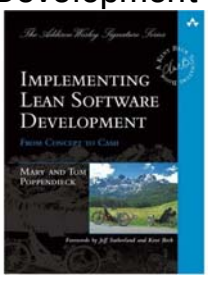
- Domain Object Modeling
- Developing by Feature
- Class (Code) Ownership
- Feature Teams
- Inspections
- Regular Build Schedule
- Configuration Management
- Visible Reporting of Results

Evo

- The first agile method? Tom Gilb,1976
- "Evolutionary Development" 1981
- "Principles of SW Engineering Management" 1988 (describes an adaptive process)
- Large influence on other agile methods

Lean Software Development

- Application of "Lean" to software
- Lean approach originated in manufacturing
- Mary and Tom Poppendieck
  - "Lean Software Development" 2003
  - "Implementing Lean SW Development" 2007



Lean Approach

- Eliminate waste
- Amplify learning, create knowledge
- Decide as late as possible, defer commitment
- Deliver as fast as possible
- Empower the team, respect people
- Build integrity in
- Optimize the whole

Agile Logic

Other Places Agile Shows Up

- Agile Unified Process
- MSF / Agile
- IBM / Open UP / Eclipse Process Framework

Agile Logic

Agile Logic
Deliver Better Software Faster

Agile Alliance

- 2001, meeting of agile methodologists
- Formed Agile Alliance
- Published Agile Manifesto
- Agile Alliance continues to run Agile 200x

Agile Logic

Agile Manifesto

- Individuals and interactions
  - Over process and tools
- Working software
  - Over comprehensive documentation
- Customer collaboration
  - Over contract negotiation
- Responding to change
  - Over following a plan
- "While there is value in the items on the right, we value the items on the left more."

Agile Logic

Agile Logic
Deliver Better Software Faster

Common Principles

- Satisfy the customer via early and frequent delivery of value
- Welcome changing requirements
- Deliver working software frequently
- Business and developers work together throughout the project
- Build project around motivated individuals
- Most effective communication is face-to-face
- Working software is primary progress measurement
- Promote sustainable development
- Be able to maintain a constant pace indefinitely
- Attention to technical excellence and good design
- Simplicity is essential
- The best architectures emerge from self-organizing teams
- Reflect at regular intervals, tune and adjust accordingly

Agile Logic
Deliver Better Software Faster

Questions?

Agile Logic
Deliver Better Software Faster