The Essence of Agile

A Tour of Agile Methods

Agile / XP So. Cal. Meeting – 1/21/09

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 Software Development Approach

- Agile Security Infrastructure
- Agile Architecture
- Agile Requirements Traceability Tool

Agile

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

What Makes Up our "Approach"?

Approach

Process
- Values
- Strategies
- Practices

Organization
- Values
- Structure
- Policies
Where Does Agile Come From?

- Ad-Hoc
- Waterfall
- Activities
- Spiral
- Iterative Incremental
- Evolutionary
- Predictive
- Adaptive
- Silo’ed
- Teams
- Lean
- Agile

Target Benefits of Agile

- Consistent, sustainable deliveries
- Ability to release in smaller increments
- Adaptation to change & risk management
- Higher release quality
- Engagement & satisfaction of customers
- Energized & positive team environment
- If combined with Lean practices...
- Efficiency & flow of feature delivery
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

Key Agile Strategies

- Collaborative Whole Teams
- Adaptive, Localized Project Management
- Evolutionary with Frequent Deliveries
- Value-Focused Clear Objectives
- Continuous Learning & Improvement
Agile Process Framework

Feedback

Guidance

Planning

Delivering

Agile Practice Areas

- **Team**
  - Collaboration
  - Cross-Functional
  - Self-Management

- **Delivery**
  - Engineering
  - Testing / TDD
  - Continuous Integration

- **Organizational**
  - Governance / Support
  - Enterprise Agile
  - Continuous Improvement

- **Product Management**
  - Product Backlog
  - Stories
  - Stakeholders

- **Project Management**
  - Release Planning
  - Iteration Planning
  - Daily Scrums
  - Velocity
  - Visibility
Agile Team Structure

Value Seekers
- Steer Project to ROI
- Stakeholders
  - End Users
  - Management
  - Marketing/Sales
  - Customer Support
  - Training/Installers
  - Technical

Builders
- Deliver Product
- Outsourced Resources
  - Tool Suppliers
- Executive Management
  - Finance
  - Human Resources
- Program Management
- Other ScrumMasters
- Enterprise Architects
  - IT / Tool Support
  - Operations
  - Other Teams
- Business Analysts
  - Product Manager
  - Testers, Architects, Tech Writers
  - Other Product Owners

Core Team
- Product Owner
- ScrumMaster
- Business Analyst
- Product Manager
- Testers, Architects, Tech Writers
- Other Product Owners
- Enterprise Architects
- IT / Tool Support
- Operations
- Other Teams
- Organizational Scrum
  - Functional Managers
  - IT Managers
  - Program Management
  - Other ScrumMasters
- Executive Management
  - Finance
  - Human Resources
- Program Management
- Other ScrumMasters

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 ???

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
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

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
Extreme Programming (XP) 1st ed.

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

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
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

XP Roles

- Separation of technical vs. business decisions
- XP roles:
  - Programmer
  - Customer
  - (Tester)
  - (Tracker)
  - (Coach)
  - (Consultant)
  - (Big Boss)
Extreme Programming (XP) 2nd ed.

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

Changes to Approach

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

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
Expanded XP Roles (Whole Team)

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

Crystal (Clear)

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

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
**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)

---

**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
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

Other Places Agile Shows Up

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

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

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.”
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

Questions?
Thank You For Attending!

Please grab some of the free stuff I brought.

Paul Hodgetts
Agile Logic
www.agilelogic.com
phodgetts@agilelogic.com
(714) 577-5795