Lean & Agile
Project Management
for Large Programs & Projects

Dr. David F. Rico, PMP, CSM

Twitter: @dr_david_f_rico
Website: http://www.davidfrico.com
LinkedIn: http://www.linkedin.com/in/davidfrico
Facebook: http://www.facebook.com/profile.php?id=1540017424
Author Background

- DoD contractor with 28+ years of IT experience
- Large gov’t projects in U.S., Far/Mid-East, & Europe

- Published six books & numerous journal articles
- Adjunct at George Washington, UMUC, & Argosy
- Agile Program Management & Lean Development
- Specializes in metrics, models, & cost engineering
- Six Sigma, CMMI, ISO 9001, DoDAF, & DoD 5000
- Cloud Computing, SOA, Web Services, FOSS, etc.
Intro to Agile Project Mgt.

Types of Agile Project Mgt.

Phases of Agile Project Mgt.

Scaling of Agile Project Mgt.

EVM for Agile Project Mgt.

Summary of Agile Project Mgt.
Today’s Whirlwind Environment

- Overruns
- Attrition
- Escalation
- Runaways
- Cancellation

Global Competition

Demanding Customers

Organization Downsizing

System Complexity

Technology Change

Vague Requirements

Work Life Imbalance
What is Agility?

- **Agility** (ə-ˈji-lə-tē) Property consisting of quickness, lightness, and ease of movement; To be very nimble
  - The ability to create and **respond to change** in order to profit in a turbulent global business environment
  - The ability to **quickly reprioritize** use of resources when requirements, technology, and knowledge shift
  - A very **fast response** to sudden market changes and emerging threats by intensive customer interaction
  - Use of **evolutionary**, **incremental**, and **iterative** delivery to converge on an optimal customer solution
  - Maximizing the **BUSINESS VALUE** with right sized, just-enough, and just-in-time processes and documentation

## Values of Agile Project Mgt.

- **People-centric** way to create innovative solutions
- **Market-centric** model to maximize business value
- **Alternative** to large document-based methodologies

### Agile Methods ‘Values’

<table>
<thead>
<tr>
<th>Customer Collaboration</th>
<th>also known as</th>
<th>Customer Interaction</th>
<th>valued more than</th>
<th>Contract Negotiation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individuals &amp; Interactions</td>
<td>also known as</td>
<td>High Performance Teams</td>
<td>valued more than</td>
<td>Processes &amp; Tools</td>
</tr>
<tr>
<td>Working Systems</td>
<td>also known as</td>
<td>Iterative Development</td>
<td>valued more than</td>
<td>Comprehensive Documentation</td>
</tr>
<tr>
<td>Responding to Change</td>
<td>also known as</td>
<td>Adaptability or Flexibility</td>
<td>valued more than</td>
<td>Following a Plan</td>
</tr>
</tbody>
</table>

---

When to use Agile Proj. Mgt.

- On exploratory or research/development projects
- When fast customer responsiveness is paramount
- In organizations that are highly innovative & creative

<table>
<thead>
<tr>
<th>Traditional Project Management</th>
<th>Agile Project Management</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Predictable situations</strong></td>
<td><strong>High levels of uncertainty and unpredictability</strong></td>
</tr>
<tr>
<td><strong>Low technology projects</strong></td>
<td><strong>High technology projects</strong></td>
</tr>
<tr>
<td><strong>Stable, slow moving industries</strong></td>
<td><strong>Fast paced, highly competitive industries</strong></td>
</tr>
<tr>
<td>Low levels of technological change</td>
<td>Rapid pace of technological change</td>
</tr>
<tr>
<td>Repeatable operations</td>
<td>Research oriented, discovery projects</td>
</tr>
<tr>
<td>Low rates of changing project performance</td>
<td>Large fluctuations in project performance</td>
</tr>
<tr>
<td>Long term, fixed price production contracts</td>
<td>Shorter term, performance based RDT&amp;E contracts</td>
</tr>
<tr>
<td>Achieving concise economic efficiency goals</td>
<td>Achieving high impact product/service effectiveness</td>
</tr>
<tr>
<td>Highly administrative contracts</td>
<td>Highly creative new product development contracts</td>
</tr>
<tr>
<td>Mass production and high volume manufacturing</td>
<td>Customer intensive, one off product/service solutions</td>
</tr>
<tr>
<td>Highly predictable and stable market conditions</td>
<td>Highly volatile and unstable market conditions</td>
</tr>
<tr>
<td>Low margin industries such as commodities</td>
<td>High margin, intellectually intensive industries</td>
</tr>
<tr>
<td>Delivering value at the point of plan</td>
<td>Delivering value at the point of sale</td>
</tr>
</tbody>
</table>

Agile World View

- “Agility” has many dimensions other than IT
- It ranges from leadership to technological agility
- The focus of this brief is program management agility
Agenda

Introduction to Agile Project Mgt.

Types of Agile Project Mgt.

Phases of Agile Project Mgt.

Scaling of Agile Project Mgt.

EVM for Agile Project Mgt.

Summary of Agile Project Mgt.
Scrum Project Management

- Created by Jeff Sutherland at Easel in 1993
- Product backlog comprised of customer needs
- Barely-sufficient project management framework

XP Project Management

- Created by Kent Beck at Chrysler in 1998
- Release plan is comprised of customer needs
- Lightweight, rigorous near-term planning element

<table>
<thead>
<tr>
<th>Exploration Phase</th>
<th>Commitment Phase</th>
<th>Steering Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build a Team</td>
<td>Sort by Value</td>
<td>Select Iteration</td>
</tr>
<tr>
<td>Write User Stories</td>
<td>Sort by Risk</td>
<td>Adjust Velocity</td>
</tr>
<tr>
<td>Estimate User Stories</td>
<td>Set Velocity</td>
<td>Insert New Stories</td>
</tr>
<tr>
<td>Split User Stories</td>
<td>Choose a Scope</td>
<td>New Release Plan</td>
</tr>
<tr>
<td>Spike User Stories</td>
<td>Set Iteration Length</td>
<td>Select Partner</td>
</tr>
<tr>
<td>Write User Tests</td>
<td>Develop Release Plan</td>
<td>Write Unit Tests</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Design and Code</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Exploration Phase</th>
<th>Commitment Phase</th>
<th>Steering Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyze Release Plan</td>
<td>Accept Tasks</td>
<td>Select Partner</td>
</tr>
<tr>
<td>Identify Iteration Goal</td>
<td>Set Individual Velocity</td>
<td>Write Unit Tests</td>
</tr>
<tr>
<td>Select User Stories</td>
<td>Estimate Tasks</td>
<td>Design and Code</td>
</tr>
<tr>
<td></td>
<td>Analyze Schedules</td>
<td>Unit/Integration Test</td>
</tr>
<tr>
<td></td>
<td>Set Load Factors</td>
<td>User Acceptance Test</td>
</tr>
<tr>
<td></td>
<td>Balance Tasks</td>
<td>Record Progress</td>
</tr>
</tbody>
</table>

Agile Project Management

- Created by Jim Highsmith at Cutter in 2003
- Focus on strategic plans and capability analysis
- Most holistic agile project management framework

Innovation Lifecycle

Envision
- Product Vision
- Product Architecture
- Project Objectives
- Project Community
- Delivery Approach

Speculate
- Gather Requirements
- Product Backlog
- Release Planning
- Risk Planning
- Cost Estimation

Explore
- Iteration Management
- Technical Practices
- Team Development
- Team Decisions
- Collaboration

Launch
- Final Review
- Final Acceptance
- Final QA
- Final Documentation
- Final Deployment

Close
- Clean Up Open Items
- Support Material
- Final Retrospective
- Final Reports
- Project Celebration

Iterative Delivery

Technical Planning
- Story Analysis
- Task Development
- Task Estimation
- Task Splitting
- Task Planning

Development, Test, & Evaluation
- Development Pairing
- Unit Test Development
- Simple Designs
- Coding and Refactoring
- Unit and Component Testing

Operational Testing
- Integration Testing
- System Testing
- Operational Testing
- Usability Testing
- Acceptance Testing

Adapt
- Focus Groups
- Technical Reviews
- Team Evaluations
- Project Reporting
- Adaptive Action

Continuous
- Standups, Architecture, Design, Build, Integration, Documentation, Change, Migration, and Integration

Story Deployment

Flexible Project Management

- Created by Doug DeCarlo at Cutter in 2004
- Focus is on collaboration, scoping, and speed
- Thinner traditional project management approach

Adaptive Project Framework

- Created by Bob Wysocki for consulting in 2008
- Designed to be a generic model for non-IT projects
- Lightweight traditional project management approach

Agenda

Intro to Agile Project Mgt.

Types of Agile Project Mgt.

Phases of Agile Project Mgt.

Scaling of Agile Project Mgt.

EVM for Agile Project Mgt.

Summary of Agile Project Mgt.
Envision Phase

- Determine product vision and project objectives
- Identifies project community and project team
- The major output is a “Product Vision Box”

Speculate Phase

- Determine organizational capability/mission needs
- Identifies feature-sets and system requirements
- The major output is a “System Release Plan”

Explore Phase

- Determine technical iteration objectives/approaches
- Identifies technical tasks and technical practices
- The major output is an “Operational Element”

Adapt Phase

- Determine the effectiveness of operational elements
- Identifies customer feedback and corrective actions
- The major output is a “Process Improvement Plan”

Close Phase

- Determine project outcome and effectiveness
- Identifies strengths, weaknesses, and rewards
- The major output is a “Lessons-Learned Report”

---

Support Material
- Finalize Documentation
- Finalize Production Material
- Finalize Manufacturing Material
- Finalize Customer Documentation
- Finalize Maintenance Information

Clean Up Open Items
- Close Open Action Items
- Close Open Change Requests
- Close Open Problem Reports
- Close Open Defect Reports
- Close Open Project Issues

Final Retrospective
- Process Performance Assessment
- Internal Product Assessment
- External Product Assessment
- Team Performance Assessment
- Project Performance Assessment

Final Reports
- End-of-Project Reports
- Administrative Reports
- Release Notes
- Financial Reports
- Facilities Reports

Project Celebration
- Individual Rewards
- Group Rewards
- Partner Rewards
- Managerial Rewards
- Product Rewards

---

Agenda

Intro to Agile Project Mgt.

Types of Agile Project Mgt.

Phases of Agile Project Mgt.

Scaling of Agile Project Mgt.

EVM for Agile Project Mgt.

Summary of Agile Project Mgt.
Multi-Level Teams

- Enables projects to plan for the future and present
- Decomposes capabilities into implementable pieces
- Unclogs the drainpipes to let the execution flow freely

![Diagram of Multi-Level Teams with roles for Product Management Team, Release Management Team, and Feature Teams]

Multi-Level Planning

- Enables multiple level enterprise plans to co-exist
- Allows stakeholders to build viewpoint-specific plans
- Ensures capabilities are delivered at regular intervals

![Diagram of Multi-Level Planning]

Multi-Level Backlog

- Enables multiple levels of abstraction to co-exist
- Allows customers and developers to communicate
- Makes optimum use of people’s time and resources

### Capabilities
- Capability 1
- Capability 2
- Capability 3

### Feature Sets
- Feature 1
- Feature 2
- Feature 3

### User Stories
- Story 1
- Story 2
- Story 3
- Story 4
- Story 5
- Story 6
- Story 7
- Story 8
- Story 9

### Multi-Level Backlog

**Feature Set**
- Cross-functional mission threads
- Related user stories that are grouped together
- Also called a Theme, i.e., implemented as an entity
- Comprises 6 to 30 days worth of work

**User Story**
- Functional, system-level requirements
- Simple requirement written by customer or user
- A small unit of functionality having business value
- Comprises 2 to 10 days worth of work

**Capability**
- Mission goal or objective level
- High-level business or product function
- Also called an Epic, i.e., multiple feature sets
- Comprises 18-90 days worth of work

Multi-Level Coordination

- Enables lean and agile methods to scale-up
- Allows enterprises to create large-scale programs
- Unleashes optimum productivity and overall control

Multi-Level Governance

- Enables enterprises to achieve functional needs
- Allows programs to coordinate functional activities
- Ensures optimal technical performance is achieved

Agenda

Intro to Agile Project Mgt.
Types of Agile Project Mgt.
Phases of Agile Project Mgt.
Scaling of Agile Project Mgt.

EVM for Agile Project Mgt.

Summary of Agile Project Mgt.
Burndown

- Most basic tracking chart for agile projects
- Tracks number of work or time units completed
- Commonly used to track no. story points completed

Cumulative Flow

- Advanced form of cumulative workflow
- Tracks planned vs. finished work or time units
- Linear progression with good project performance

Agile EVM

- Adaptation of EVM for agile projects
- Mapping between traditional and agile projects
- Work completed is more authoritative in agile projects

Earned Business Value

- ROI is estimated for user stories in agile projects
- Value accrues with each completed user story
- Value of completed tasks is more meaningful

Agenda

Intro to Agile Project Mgt.
Types of Agile Project Mgt.
Phases of Agile Project Mgt.
Scaling of Agile Project Mgt.
EVM for Agile Project Mgt.

Summary of Agile Project Mgt.
Case Studies

- 70% of worldwide IT projects use agile methods
- Includes highly-regulated industries like U.S. DoD
- Even split between top-down and bottom-up adoption

<table>
<thead>
<tr>
<th>Industry</th>
<th>Org</th>
<th>Project</th>
<th>Purpose</th>
<th>Size</th>
<th>Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electronic Commerce</td>
<td>Google</td>
<td>Adwords</td>
<td>Advertising</td>
<td>• 20 teams</td>
<td>• 1,838 User Stories</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 140 people</td>
<td>• 6,250 Function Points</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 5 countries</td>
<td>• 500,000 Lines of Code</td>
</tr>
<tr>
<td>Shrink Wrapped</td>
<td>Primavera</td>
<td>Primavera</td>
<td>Project Management</td>
<td>• 15 teams</td>
<td>• 26,809 User Stories</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 90 people</td>
<td>• 91,146 Function Points</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Collocated</td>
<td>• 7,291,666 Lines of Code</td>
</tr>
<tr>
<td>Health Care</td>
<td>FDA</td>
<td>m2000</td>
<td>Blood Analysis</td>
<td>• 4 teams</td>
<td>• 1,659 User Stories</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 20 people</td>
<td>• 5,640 Function Points</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Collocated</td>
<td>• 451,235 Lines of Code</td>
</tr>
<tr>
<td>Law Enforcement</td>
<td>FBI</td>
<td>Sentinel</td>
<td>Case File Workflow</td>
<td>• 10 teams</td>
<td>• 3,947 User Stories</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 50 people</td>
<td>• 13,419 Function Points</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Collocated</td>
<td>• 1,073,529 Lines of Code</td>
</tr>
<tr>
<td>U.S. DoD</td>
<td>Stratcom</td>
<td>SKIweb</td>
<td>Knowledge Management</td>
<td>• 3 teams</td>
<td>• 390 User Stories</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 12 people</td>
<td>• 1,324 Function Points</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Collocated</td>
<td>• 105,958 Lines of Code</td>
</tr>
</tbody>
</table>

Leadership Considerations

- Agile management is delegated to the lowest level
- There remain key leadership roles & responsibilities
- Communication, coaching, & facilitation are key ones

<table>
<thead>
<tr>
<th>Customer Communication</th>
<th>Facilitate selection of methods for obtaining and maintaining executive commitment, project resources, corporate communications, and customer interaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Visioning</td>
<td>Facilitate selection of methods for communicating product purpose, goals, objectives, mission, vision, business value, scope, performance, budget, assumptions, constraints, etc.</td>
</tr>
<tr>
<td>Distribution Strategy</td>
<td>Facilitate selection of virtual team distribution strategy to satisfy project goals and objectives</td>
</tr>
<tr>
<td>Team Development</td>
<td>Facilitate selection of methods for training, coaching, mentoring, and other team building approaches</td>
</tr>
<tr>
<td>Standards &amp; Practices</td>
<td>Facilitate selection of project management and technical practices, conventions, roles, responsibilities, and performance measures</td>
</tr>
<tr>
<td>Telecom Infrastructure</td>
<td>Facilitate selection of high bandwidth telecommunication products and services</td>
</tr>
<tr>
<td>Development Tools</td>
<td>Facilitate selection of agile project management tools and interactive development environment</td>
</tr>
<tr>
<td>High Context Meetings</td>
<td>Facilitate selection of high context agile project management and development meetings</td>
</tr>
<tr>
<td>Coordination Meetings</td>
<td>Facilitate selection of meetings and forums for regular communications between site coordinators</td>
</tr>
<tr>
<td>F2F Communications</td>
<td>Facilitate selection of methods for maximizing periodic face to face interactions and collaboration</td>
</tr>
<tr>
<td>Performance Management</td>
<td>Facilitates selection of methods for process improvement, problem resolution, conflict management, team recognition, product performance, and customer satisfaction</td>
</tr>
</tbody>
</table>

## Advanced Agile Measures

- Agile Methods are a fundamentally new paradigm
- Agile Methods are “not” lighter Traditional Methods
- They should not be viewed through a traditional lens

<table>
<thead>
<tr>
<th>Agile Metrics</th>
<th>Traditional Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Customer Collaboration</strong></td>
<td><strong>Contracts</strong></td>
</tr>
<tr>
<td>- Interaction frequency</td>
<td>- Contract compliance</td>
</tr>
<tr>
<td>- Comm. quality</td>
<td>- Contract deliverables</td>
</tr>
<tr>
<td>- Relationship strength</td>
<td>- Contract change orders</td>
</tr>
<tr>
<td><strong>Individuals &amp; Interactions</strong></td>
<td><strong>Processes</strong></td>
</tr>
<tr>
<td>- Team competence</td>
<td>- Lifecycle compliance</td>
</tr>
<tr>
<td>- Team motivation</td>
<td>- Process Maturity Level</td>
</tr>
<tr>
<td>- Team cooperation</td>
<td>- Regulatory compliance</td>
</tr>
<tr>
<td><strong>Working Software</strong></td>
<td><strong>Documentation</strong></td>
</tr>
<tr>
<td>- Iteration size</td>
<td>- Document deliveries</td>
</tr>
<tr>
<td>- Iteration length</td>
<td>- Document comments</td>
</tr>
<tr>
<td>- Iteration number</td>
<td>- Document compliance</td>
</tr>
<tr>
<td><strong>Responding to Change</strong></td>
<td><strong>Project Plans</strong></td>
</tr>
<tr>
<td>- Org. flexibility</td>
<td>- Cost Compliance</td>
</tr>
<tr>
<td>- Mgt. flexibility</td>
<td>- Scope Compliance</td>
</tr>
<tr>
<td>- Individual flexibility</td>
<td>- Schedule Compliance</td>
</tr>
</tbody>
</table>

Organizational Change

- Change, no matter how small or large, is difficult
- Smaller focused changes help to cross the chasm
- Shrinking, simplifying, and motivation are key factors

How to Cross the Chasm

Switch: How to Change Things When Change is Hard

Direct the Rider
- Follow the bright spots - Clone what works
- Script the critical moves - Use prescriptive behaviors
- Point to the destination - Focus on the end game

Motivate the Elephant
- Find the feeling - Appeal to emotion
- Shrink the change - Use incremental change
- Grow your people - Invest in training and education

Shape the Path
- Tweak the environment - Simplify the change
- Build habits - Create simple recipes for action
- Rally the herd - Get everyone involved

Influencer: The Power to Change Anything

Make the Undesirable Desirable
- Create new experiences - Make it interesting
- Create new motives - Appeal to sensibility

Surpass your Limits
- Perfect complex skills - Establish milestones
- Build emotional skills - Build maturity and people skills

Harness Peer Pressure
- Recruit public personalities - Involve public figures
- Recruit influential leaders - Involve recognized figures

Find Strength in Numbers
- Utilize teamwork - Enlist others to help out
- Enlist the power of social capital - Scale up and out

Design Rewards and Demand Accountability
- Use incentives wisely - Reward vital behaviors
- Use punishment sparingly - Warn before taking action

Change the Environment
- Make it easy - Simplify the change
- Make it unavoidable - Build change into daily routine

New contract models emerged for agile contracts
Goals, objectives, and visions are established early
Buyers and suppliers collaborate throughout contract

<table>
<thead>
<tr>
<th>Contract Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic Value</td>
<td>Specify initial scope and needs (with iterative enhancements)</td>
</tr>
<tr>
<td>Performance Based</td>
<td>Establish performance objectives (but not technical solutions)</td>
</tr>
<tr>
<td>Target Cost</td>
<td>Broad boundaries for time, cost, and quality (but not scope)</td>
</tr>
<tr>
<td>Optional Scope</td>
<td>Set minimum and maximum costs (based on initial scope)</td>
</tr>
<tr>
<td>Collaborative</td>
<td>Outline initial scope (with fixed no. of releases and iterations)</td>
</tr>
<tr>
<td>Lean</td>
<td>Lean tools such as small batches, Kanban, WIP constraints, etc.</td>
</tr>
</tbody>
</table>

How do Lean & Agile Intersect?

- Agile is **naturally** lean and based on small batches
- Agile directly **supports** six principles of lean thinking
- Agile may be **converted** to a continuous flow system

<table>
<thead>
<tr>
<th>Agile Values</th>
<th>Lean Pillars</th>
<th>Lean Principles</th>
<th>Lean &amp; Agile Practices</th>
<th>Flow Principles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Empowered Teams</td>
<td>Respect for People</td>
<td>Relationships</td>
<td>• Customer relationships, satisfaction, trust, and loyalty&lt;br&gt;• Team authority, empowerment, and resources&lt;br&gt;• Team identification, cohesion, and communication</td>
<td>Decentralization</td>
</tr>
<tr>
<td>Customer Collaboration</td>
<td></td>
<td>Customer Value</td>
<td>• Product vision, mission, needs, and capabilities&lt;br&gt;• Product scope, constraints, and business value&lt;br&gt;• Product objectives, specifications, and performance</td>
<td>Economic View</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Value Stream</td>
<td>• As is policies, processes, procedures, and instructions&lt;br&gt;• To be business processes, flowcharts, and swim lanes&lt;br&gt;• Initial workflow analysis, metrcication, and optimization</td>
<td>WIP Constraints &amp; Kanban</td>
</tr>
<tr>
<td>Iterative Delivery</td>
<td>Continuous Flow</td>
<td>Continuous Flow</td>
<td>• Batch size, work in process, and artifact size constraints&lt;br&gt;• Cadence, queue size, buffers, slack, and bottlenecks&lt;br&gt;• Workflow, test, integration, and deployment automation</td>
<td>Control Cadence &amp; Small Batches</td>
</tr>
<tr>
<td>Responding to Change</td>
<td></td>
<td>Customer Pull</td>
<td>• Roadmaps, releases, iterations, and product priorities&lt;br&gt;• Epics, themes, feature sets, features, and user stories&lt;br&gt;• Product demonstrations, feedback, and new backlogs</td>
<td>Fast Feedback</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Perfection</td>
<td>• Refactor, test driven design, and continuous integration&lt;br&gt;• Standups, retrospectives, and process improvements&lt;br&gt;• Organization, project, and process adaptability/flexibility</td>
<td>Manage Queues/Exploit Variability</td>
</tr>
</tbody>
</table>

Conclusion

- Agility is the evolution of management thought
- Confluence of traditional and non-traditional ideas
- Improve performance by over an order-of-magnitude

“Don’t waste your time using traditional project management on 21st century projects”

APM Textbooks

- Over 15 text books for agile project management
- Many of them stem from Planning XP by Kent Beck
- Agile Project Mgt. by Jim Highsmith is most complete