Decomposing Large Programs into Agile Projects &
Mapping PMI Processes onto Agile Best Practices

Presented by:
Mike Griffiths, PMP
Leading Answers
May 15, 2008

2007/2008 Calgary APLN Topics

October Planning meeting, top ranked topics:

1. Using Agile on distributed teams
2. Team collaboration / motivation / accountability
3. Decomposing large programs into Agile projects
4. Mapping PMI processes onto Agile best practices
5. Fitting agile into the constraints imposed by the business
2007/2008 Calgary APLN Topics

October Planning meeting, top ranked topics:

1. Using Agile on distributed teams ✓
2. Team collaboration / motivation / accountability ✓
3. Decomposing large programs into Agile projects
4. Mapping PMI processes onto Agile best practices
5. Fitting agile into the constraints imposed by the business – June 18 case study

Agenda
Why do large Programs/Projects fail?

Top 5 Reasons:
1. Complexity
2. Business Change
3. Personnel Change
4. Communication Challenges
5. Compounding Uncertainties
Why do large Programs/Projects fail?

1. Complexity increase is non-linear
   COCOMO Effort = Size^{Penalty} * Productivity Factor * Productivity Adjusters

2. Business change duration risks
   - Rates of change
   - Globalization
   - Future Requirements

3. People change duration risks
   - Changing priorities
   - Changing sponsors
   - Changing team members
Why do large Programs/Projects fail?

4. Communication channels increase exponentially. CC = \frac{n(n-1)}{2}

Why do large Programs/Projects fail?

5. Compounding Task Completion probabilities get small fast

<table>
<thead>
<tr>
<th>Number of 95% Tasks</th>
<th>Probability of completing to plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>60%</td>
</tr>
<tr>
<td>50</td>
<td>28%</td>
</tr>
<tr>
<td>100</td>
<td>0.6%</td>
</tr>
<tr>
<td>200</td>
<td>0.003%</td>
</tr>
</tbody>
</table>
Chopping Projects Down to Size

Reduction Strategies:
1. **By Functional area**
   - E.G. capture customer orders, manage inventory, and produce financial reports

2. **Plain-Vanilla First**
   - 80% of transactions that are straightforward
   - Add exceptions as a follow on project
   - Balance open design against YAGNI

3. **By User Group**
   - E.G. Order Entry users first
   - Management reporting as a later project
   - (Need to ensure necessary details are captured in the first project)

4. **By Sub-system Replacements**
   - Assess the architecture then tackle them one at a time

Options to avoid include:

1. **By Project Phase**
   - Requirements Gathering and Analysis as one project
   - Develop and Test as another project

2. **By Financial Year**
   - Not really smaller
   - Duration risks still present
Synchronizing Multiple Projects

Mapping PMI to Agile Outline
From Amazon: "The authors begin by explaining how agile works... next, they systematically map the PMI's classic, methodology-independent techniques and terminology to agile practices..."
Different Lifecycle Focus

Based on Methodology Scope diagrams from Alistair Cockburn
The PMI Process Groups
### PMI Knowledge Areas

#### Project Management

4. **Project Integration Management**
   - 4.1 Project Plan Development
   - 4.2 Project Plan Execution
   - 4.3 Integrated Change Control

5. **Project Scope Management**
   - 5.1 Initiation
   - 5.2 Scope Planning
   - 5.3 Scope Definition
   - 5.4 Scope Verification
   - 5.5 Scope Change Control

6. **Project Time Management**
   - 6.1 Activity Definition
   - 6.2 Activity Sequencing
   - 6.3 Activity Duration Estimating
   - 6.4 Schedule Development
   - 6.5 Schedule Control

7. **Project Cost Management**
   - 7.1 Resource Planning
   - 7.2 Cost Estimating
   - 7.3 Cost Budgeting
   - 7.4 Cost Control

8. **Project Quality Management**
   - 8.1 Quality Planning
   - 8.2 Quality Assurance
   - 8.3 Quality Control

9. **Project Human Resource Management**
   - 9.1 Organizational Planning
   - 9.2 Staff Acquisition
   - 9.3 Team Development

10. **Project Communications Management**
    - 10.1 Communications Planning
    - 10.2 Information Distribution
    - 10.3 Performance Reporting
    - 10.4 Administrative Closure

11. **Project Risk Management**
    - 11.1 Risk Management Planning
    - 11.2 Risk Identification
    - 11.3 Qualitative Risk Analysis
    - 11.4 Quantitative Risk Analysis
    - 11.5 Risk Response Planning
    - 11.6 Risk Monitoring and Control

12. **Project Procurement Management**
    - 12.1 Procurement Planning
    - 12.2 Solicitation Planning
    - 12.3 Solicitation
    - 12.4 Source Selection
    - 12.5 Contract Administration
    - 12.6 Contract Closeout

---

#### Process Groups

<table>
<thead>
<tr>
<th>Knowledge Area</th>
<th>Initiating</th>
<th>Planning</th>
<th>Executing</th>
<th>Monitoring and Controlling</th>
<th>Closing</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Project Integration Management</td>
<td>4.1 Project Plan Development</td>
<td>4.2 Project Plan Execution</td>
<td>4.3 Integrated Change Control</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Project Scope Management</td>
<td>5.1 Initiation</td>
<td>5.2 Scope Planning</td>
<td>5.3 Scope Definition</td>
<td>5.4 Scope Verification</td>
<td>5.5 Scope Change Control</td>
</tr>
<tr>
<td>6. Project Time Management</td>
<td>6.1 Activity Definition</td>
<td>6.2 Activity Sequencing</td>
<td>6.3 Activity Duration Estimating</td>
<td>6.4 Schedule Development</td>
<td>6.5 Schedule Control</td>
</tr>
<tr>
<td>8. Project Quality Management</td>
<td>8.1 Quality Planning</td>
<td>8.2 Quality Assurance</td>
<td>8.3 Quality Control</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9. Project Human Resource Management</td>
<td>9.1 Organizational Planning</td>
<td>9.2 Staff Acquisition</td>
<td>9.3 Team Development</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10. Project Communications Management</td>
<td>10.1 Communications Planning</td>
<td>10.2 Information Distribution</td>
<td>10.3 Performance Reporting</td>
<td>10.4 Administrative Closure</td>
<td></td>
</tr>
</tbody>
</table>

*Agile Project Management Processes Iterate...*
Agile Best Practices Iterate

Agile Project Management Lifecycle

Figure 3-2. Overlap of Process Groups in a Phase

From: PMBOK v4 Exposure Draft
PMI to Agile Best Practice Mapping

Agile Principles:
1. Business satisfaction is primary goal
2. Deliver working software frequently
3. Welcome changing requirements
4. Business and team work together throughout the project
5. Motivate, support and trust teams to get the job done
6. Focus on face-to-face communications
7. Working software is the primary measure of progress
8. Promote a sustainable pace
9. Continuous attention to technical excellence
10. Simplify wherever possible
11. Promote self-organizing teams
12. Encourage regular reflection and improvement

PMI Mappings:
- Should be, (on time, budget, to spec. can override)
- Could do, but not usually a priority
- Processes tend to discourage changes
- Possible, but not part of process
- Yes, as part of 9.3 Team Development
- Could do, but not an emphasis of Comms. Mgmt
- Not normally, progress against plan
- Should be as part of 4.2 Project Plan Execution
- Could do, but processes can lead to high complexity
- Not really, command-and-control, task dispatching
- Could do, but guidelines for LL at project end

Summary

Large programs/projects risky, small projects less risky
If you have large programs/projects try to break them down

PMI and Agile methods have same target destination (successful projects)
PMI takes the "plan-the-work, work-the-plan" route
Agile takes the "collaboration with business and empowered teams" route

The PMI method works well for defined scope projects with few uncertainties
Agile methods work well for hard to define projects with many uncertainties

The PMI method has a strong process emphasis
Agile methods have a strong people emphasis

You can augment one approach with tools from the other, but
the core choice should be made on project and organizational factors
Book Offer

SAVE 35%!
Just order the print book or PDF electronic version from www.informIT.com, and enter the coupon code SLIGER2759 at checkout.

Michele Sliger
Stacia Broderick

The Software Project Manager's Bridge to Agility

When software development teams move to agile methods, experienced project managers often struggle — doubtful about the new approach and uncertain about their new roles and responsibilities. In this book, two long-time certified Project Management Professionals (PMPs) and Scrum trainers have built a bridge to this dynamic new paradigm. They show experienced project managers how to successfully transition to agile by refocusing on facilitation and collaboration, not "command and control."

The authors begin by explaining how agile works; how it differs from traditional "plan-driven" methodologies, the benefits it promises, and the real-world results it delivers. Next, they systematically map the Project Management Institute's classic, methodology-independent techniques and terminology to agile practices. They cover both process and project lifecycles and carefully address vital issues ranging from scope and time to