Agile Development and Its Impact on Productivity

David Garmus
Agenda

- Characteristics of an Agile project
- Performance data on Agile projects
- When to choose Agile
Characteristics of an Agile Project

- Characteristics of an Agile project
  - Common view
  - Core practices
  - Lifecycle development

- View of an Agile project: Rigid methods have one thing in common: too much is planned in an uncertain environment at project inception.

- “XP” (the most common Agile methodology) is a lightweight methodology for small-to-medium-sized teams developing software in the face of vague or rapidly changing requirements.” – Kent Beck
## Characteristics Overview

<table>
<thead>
<tr>
<th>Barry Boehm and Richard Turner</th>
<th>Agile</th>
<th>Planned (Traditional)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Changeable</td>
<td>Larger Teams &amp; Projects</td>
</tr>
<tr>
<td>Management</td>
<td>Customer is Part of Team</td>
<td>Document Plans &amp; Requirements</td>
</tr>
<tr>
<td>Technical</td>
<td>Short Increments; Many Releases</td>
<td>Voluminous Test Cases and Plans</td>
</tr>
<tr>
<td>Personnel</td>
<td>Highly Capable; Thrive on Chaos</td>
<td>Specialists Who Thrive on Order</td>
</tr>
</tbody>
</table>
Core Practices

• Delivery of working software is the primary goal
• Active stakeholder participation
• Assimilate change easily
  – Incremental approach
  – Use of simple models to address requirements
  – Rapid feedback
  – Frequent deliverables
  – Flexible change management
• Steady development rate
• Quality through teamwork
• Direct communication
• Maximize agility through model normalization
• Retain key models for reuse
Lifecycle Development: Requirements, Design and Coding

• Small in scope
  – Less than 200 Function Points
  – Many less than 100 Function Points
• High-level business requirements
• Lack of formal documentation
  – System descriptions
  – Technical design documentation
  – Process models
  – System architectures
• Team development
  – Small, cohesive, same work space
  – Self organized and motivated
  – Driven by team dynamics vice formal process
Lifecycle Development: Testing and Implementation

• Effort not separate for each testing type/phase
• Testing occurs concurrently
• Testing by developers
• Smaller deliverables, more releases
Agenda

- Characteristics of an Agile project
- Performance data on Agile projects
- When to choose Agile
Use Measurement to Enable Comparisons Between Methods

QUANTITATIVE
- Deliverable Size
- Effort/Cost
- Duration
- Quality

QUALITATIVE
- Process
- Methods
- Skills
- Tools
- Environment

Measure how you are doing

Measured Performance

Capable Maturity

Standard of performance

Baseline of Performance

Identify what you are doing
Utilize Measurement Results in Decision Making

- Improvements resulting from current and future initiatives must be measured
- The basis for measuring improvements may include:
  - Industry data
  - Organizational baseline data

It is necessary for the organization to put a “stake in the ground” relative to current performance level in order to improve development practices.
### Characteristics
- Project Type
- Platform
- Data Base
- Method
- Language

### Complexity Variables
- Logical Algorithms
- Mathematical Algorithms
- Data Relationships
- Functional Size
- Reuse
- Code Structure
- Performance
- Memory
- Security
- Warranty

### Metrics
- Size
- Cost
- Effort
- Duration
- Defects

### Attributes
- Management
- Definition
- Design
- Build
- Test
- Environment
- Process
- Skill Levels
- Quality Practices
- Measures
Collecting and Reporting

- Identify data set (typically project oriented)
- Collect baseline data
  - Project measures (e.g., effort, size, cost, duration, defects)
  - Project attributes (e.g., skill levels, tools, process, etc.)
- Analyze data
  - Performance comparisons (identification of process strengths and weaknesses)
  - Industry averages and best practices
  - Performance modeling (identify high impact areas)
- Report results
# Hours per Function Point

Average Hours / Function Point of Recent Enhancement Projects Across Different Platforms from DCG Database for Small Projects

<table>
<thead>
<tr>
<th></th>
<th>Traditional</th>
<th>Agile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Server</td>
<td>6.5</td>
<td>4.2</td>
</tr>
<tr>
<td>Main Frame</td>
<td>8.1</td>
<td>7.0</td>
</tr>
<tr>
<td>Web</td>
<td>4.8</td>
<td>3.2</td>
</tr>
<tr>
<td>E-business Web</td>
<td>6.6</td>
<td>5.8</td>
</tr>
</tbody>
</table>
Using Historical Delivery Rates

DEFINITION

PROJECT SIZE and COMPLEXITY

FUNCTION POINT SIZE

CAPABILITY

RATE OF DELIVERY

HOURS per FUNCTION POINT

EFFORT

Schedule

Effort

Costs
Agenda

• Characteristics of an Agile project

• Performance data on Agile projects

• When to choose Agile
Which Methodology Should I Use?

**Agile Methods**
- Exploratory projects
- Small teams
- Participative environments
  - Experienced personnel
  - Active business partners
- Software dominant projects
- In-sourced projects
- High risk of unknown requirements

**Waterfall, Iterative and Spiral Methods**
- Predictive performance
- Large teams
- Highly structured environments
- Outsourced or multi-sourced projects
- High financial or safety risk
- Significant hardware integration
Do I Choose Agile or Hybrid?

• Initial arguments for selecting a hybrid (traditional) method:
  – High level of risk
  – Large size of project
  – Specified delivery commitment
  – Organizational environment

• Suggested selection process:
  – Map Agile attributes based on organization’s tolerance for risk and change
  – Some Agile practices can be transplanted to another methodology
  – Leverage best practice processes to augment method chosen
Key Attributes of Agile Projects

• The planning game
• On-site customer / user
• System metaphor
• Simple design
• Collective code ownership
• Coding standards
• Pair programming
• Continuous testing
• Continuous integration
• Small releases
• 40 hour work week

Key Attributes of Current Agile Methods

➢ Well defined
➢ Require significant discipline
Example: Team Size as a Driver for Hybrid

<table>
<thead>
<tr>
<th>Small Team</th>
<th>Medium Team</th>
<th>Large Team</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 People</td>
<td>12 People</td>
<td>13 or More</td>
</tr>
</tbody>
</table>

The planning game
On-site customer / user
System metaphor
Simple design
Collective code ownership
Coding standards
Pair programming
Continuous testing
Continuous Integration
Small releases
40 hour work week
Example: Financial Risk as a Driver for Hybrid

Low Financial Risk

No Impact to Accounts

Large Financial Risk

Impact to Accounts

The planning game
On-site customer / user
System metaphor
Simple design
Collective code ownership
Coding standards
Pair programming
Continuous testing
Continuous integration
Small releases
40 hour work week
Summary

• Use of Agile methods affects performance outcomes

• Choosing the appropriate methodology will maximize your delivery performance

• Agile performance can be successful
Contact Us

Email: info@softwarevalue.com

Phone: 1-610-644-2856

http://www.softwarevalue.com

@DCGSWValue
/DCGSoftwareValue
/company/DCG-Software-Value