Discipline vs. Agility

EECS810: Software Engineering

EECS811: SW Project Management
Topics

• What is discipline?
• What is agility?
• What are the misconceptions?
• Contrasts and home grounds
• Five critical factors
Where did discipline come from?

- DoD guidance documents
  - MIL-STD-1521
  - DoD-STD-2167
  - MIL-STD-498

- Large commercial companies
  - IBM
  - Hitachi
  - Siemens
What is disciplined?

- Adjectives
  - Predictive
  - Plan-driven
  - Documentation
  - Systematic
Related concepts

- Process improvement
- Process capability
- Organizational maturity
- Process group
- Risk management
- Verification
- Validation
Topics

• What is discipline?
• What is agility?
• What are the misconceptions?
• Contrasts and home grounds
• Five critical factors
Where did agility come from?

- Outgrowth of rapid prototyping
- Programming more of a craft than a process
- Address a common problem: After a long development cycle the product doesn’t meet expectations
What is agile? [1/3]

- Agile manifesto: We value
  - Individuals and interactions over process and tools
  - Working software over comprehensive documentation
  - Customer collaboration over following a plan
  - That is, while there is value in the items on the right, we value the items on the left more
What is agile? [2/3]

- **Adjectives:**
  - Iterations
  - Test-driven
  - Customer collaboration

- **Methods:**
  - eXtreme Programming (XP)
  - Adaptive Software Development
  - Feature Driven Development
  - Scrum

Every 24 hours
What is agile? [3/3]

• Embrace change
• Fast cycle/frequent delivery
• Simple design
• Refactoring
• Pair programming
• Retrospective
• Test-driven development
Sounds great, why not use it?

- Agile has trouble scaling
  - Size of project
  - Size of group
- Cost can go up with group size
- Plan-driven has trouble trimming
  - Heavy documentation
  - Late cycle delivery
- No silver bullet
What are the key differences?

- Plan-driven models value process over people; agile models value people over process
- Document, document, document - chants the disciplined
Topics

• What is discipline?
• What is agility?
• **What are the misconceptions?**
• Contrasts and home grounds
• Five critical factors
What are the misconceptions?

<table>
<thead>
<tr>
<th>Plan-Driven Methods</th>
<th>Agile Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plan-driven methods are uniformly bureaucratic.</td>
<td>Agile methods do not plan.</td>
</tr>
<tr>
<td>Having documented plans guarantees compliance with plans.</td>
<td>Agile methods require uniformly talented people.</td>
</tr>
<tr>
<td>Plan-driven methods can succeed with a lack of talented people.</td>
<td>Agile methods can make the slope of the cost-to-change vs. time curve uniformly flat.</td>
</tr>
<tr>
<td>High maturity guarantees success.</td>
<td>YAGNI is a universally safe assumption and will not alienate your customers.</td>
</tr>
</tbody>
</table>
Topics

• What is discipline?
• What is agility?
• What are the misconceptions?
• Contrasts and home grounds
• Five critical factors
Application characteristics contrasts and home grounds

• Primary goals
  - Agile goals are rapid value and responsiveness to change
  - Plan-driven goals are predictability, stability, and high assurance

• Size
  - Agile works best on smaller projects
  - Plan-driven is a necessity on large complex projects

• Environment
  - Agile approaches are comfortable in high-change environments with some risks
  - Plan-driven methods need stability
Management characteristics contrasts and home grounds

- **Customer relations**
  - Agile encourages a dedicated *collocated* customer
  - Plan-driven methods depend on *contracts and specifications*

- **Planning and control**
  - Agilists see planning as a means to an end
  - Plan-driven methods use plans to *communicate and coordinate*

- **Project communication**
  - Agile methods depend on *tacit* knowledge
  - Plan-driven approaches use *explicit, documented knowledge*
Technical Characteristics
Contrasts and Home Grounds

- Requirements
  - Agile uses informal, user-prioritized stories as requirements
  - Plan-driven methods prefer specific, formalized requirements

- Development
  - Agile advocates simple design
  - Plan-driven methods advocate architecture to anticipate changes

- Testing
  - Agile methods develop tests before code, and test incrementally
  - Plan-driven methods test to specifications
Personnel characteristics contrasts and home grounds

- **Customers**
  - Both methods need collaborative, authorized, committed, and knowledgeable representative
  - Plan-driven does not require them full-time

- **Developers**
  - Agile developers need more than technical skills
  - Plan-driven methods need fewer highly talented people than agile

- **Culture**
  - Agilists like many degrees of freedom
  - Plan-driven people need clear process and roles
Topics

- What is discipline?
- What is agility?
- What are the misconceptions?
- Contrasts and home grounds
- **Five critical factors**
Five Critical Factors

- Factors to measure:
  - Personnel
  - Size
  - Dynamism
  - Criticality
  - Culture
Five critical factors: personnel

• Agile
  - Requires continuous presence of critical mass of scarce Cockburn Level 2 or 3 experts; risky to use non-agile Level 1 people.

• Plan-driven
  - Needs a critical mass of level 2 and level 3 experts during project definition, but can work with fewer later in the project - unless the environment is highly dynamic; can usually accommodate some Level 1 people
# Cockburn Scale

<table>
<thead>
<tr>
<th>Level</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Able to revise in unprecedented situation</td>
</tr>
<tr>
<td>2</td>
<td>Able to tailor a method to fit new situation</td>
</tr>
<tr>
<td>1A</td>
<td>With training can perform discretionary steps, can train to level 2</td>
</tr>
<tr>
<td>1B</td>
<td>With training can perform basic procedural steps</td>
</tr>
<tr>
<td>-1</td>
<td>May have technical skills but unable or unwilling to collaborate</td>
</tr>
</tbody>
</table>
Five critical factors: size

- **Agile**
  - Well matched to **small** products and teams

- **Plan-driven**
  - Methods evolved to handle **large** products and teams
Five critical factors: dynamism

- Agile
  - Simple design and continuous refactoring are excellent for highly dynamic environments, but a source of potentially expensive rework for highly stable environments.

- Plan-Driven
  - Detailed plans and big design up front excellent for highly stable environments, but a source of expensive rework for highly dynamic environments.
Five critical factors: criticality

- **Agile**
  - Untested on safety-critical products
  - Open to change in requirements

- **Plan-driven**
  - Methods evolved to handle highly critical products
  - Not open to change in requirements
Five critical factors: culture

• Agile
  - Thrives in a culture where people feel comfortable and empowered by having many degrees of freedom

• Plan-driven
  - Thrives in a culture where people feel comfortable and empowered by having their roles defined by clear policies and procedures
Walking the line: When to use one
An agile methodology is preferred
A disciplined methodology is preferred