It Depends
On
Common Sense

www.controlchaos.com
www.agilealliance.org
Ken.schwaber@verizon.net
Agile Framework

Daily Status

Iteration

Product Backlog Stories

Increment of Functionality
Let people figure out the right thing to do, and then do it.

Let people be creative.

OOPSLA'02
Scaling Agile Processes

“Scaling Agile projects is the last thing you should do.”
Martin Fowler

<table>
<thead>
<tr>
<th>1m lines of C++ code</th>
<th>BQP</th>
<th>Industry Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time in months</td>
<td>31</td>
<td>&gt;50</td>
</tr>
<tr>
<td>Staff</td>
<td>8</td>
<td>&gt;100</td>
</tr>
<tr>
<td>Function Points</td>
<td>77</td>
<td>2</td>
</tr>
</tbody>
</table>

“So in the extreme case for large projects, it is possible to scale to a 500 developer effort with only 40 developers on staff and deliver the project in almost half the time with SCRUM.”
Jeff Sutherland
How Agile Methodologies Scale

Staging the Product Backlog

Figure 3
How Agile Methodologies Scale

Initial Product Backlog
- Functional requirement
- Non-functional requirements
- Staged, scalability requirements
- the rest of the functional and non-functional requirements

Product Backlog
- Functional requirements
- Non-functional requirements

Single Team

Many Teams
Large Project, Multiple Teams

1. For smaller projects the vision statement coupled with emergence and refactoring is adequate.

2. Some architecture and design needed when multiple teams are used.
Maximized Control Through Inspection and Adaptation

- Defined
- Empirical

Flexible response to unpredictability improves $p(\text{Success})$ to Complexity relationship

Increased probability($\text{success}$)

Probability of Success

Low Complexity | Medium Complexity | High Complexity

Edge of Chaos
How Scrum Works

1. Initiation
   - Staffing
   - Educating

2. Planning
   - New Unfunded Projects
   - New Funded Projects
   - Ongoing Projects
   - Fixed Price/Fixed Data Projects

3. Type of Development
   - Internal Software
   - Commercial Software
   - FDA Life Critical Software
   - Mission and Timing Critical Software
   - Package and COTS Development

4. Type of Project
   - Multiple Teams
   - Single Team
   - Complex Architecture System

5. Development Environment Enhancements

6. Development
   - Sprints
   - Product Increment

7. Release
   - Sprints
# Scrum Compliance with CMM Software Framework

<table>
<thead>
<tr>
<th>Level</th>
<th>Key Practice Area</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Requirements management</td>
<td>√√</td>
</tr>
<tr>
<td>2</td>
<td>Software project planning</td>
<td>√√</td>
</tr>
<tr>
<td>2</td>
<td>Software project tracking and oversight</td>
<td>√√</td>
</tr>
<tr>
<td>2</td>
<td>Software subcontract management</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Software quality assurance</td>
<td>√√</td>
</tr>
<tr>
<td>2</td>
<td>Software configuration management</td>
<td>√</td>
</tr>
<tr>
<td>3</td>
<td>Organization process focus</td>
<td>√</td>
</tr>
<tr>
<td>3</td>
<td>Organization process definition</td>
<td>√</td>
</tr>
<tr>
<td>3</td>
<td>Training program</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Integrated software management</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Software product engineering</td>
<td>√√</td>
</tr>
<tr>
<td>3</td>
<td>Intergroup coordination</td>
<td>√√</td>
</tr>
<tr>
<td>3</td>
<td>Peer review</td>
<td></td>
</tr>
</tbody>
</table>
Right brain

Over

Left brain