Introduction to Disciplined Agile Delivery (DAD)

Mark Lines  
Calgary, Alberta, Canada  
Co-founder, UPMentors  
Mark@UPMentors.com  
www.DisciplinedAgileDelivery.com  
twitter.com/Mark_Lines
Agenda

- Defining Disciplined Agile Delivery (DAD)
- People first
- Learning oriented
- Hybrid agile framework
- A risk-value driven lifecycle
- Goals driven lifecycle
  - Inception
  - Construction
  - Transition
- Enterprise aware
  - Optimize the whole
  - Agile governance
- Agility@scale
- Questions
Defining DAD

- The Disciplined Agile Delivery (DAD) process framework is a hybrid agile approach to IT solution delivery that is:
  - People-first
  - Learning-oriented
  - Risk and value driven
  - Goal-driven
  - Scalable
  - Enterprise aware
People First: Potential roles on DAD teams

- **Primary roles:**
  - Stakeholder
  - Team Lead
  - Product Owner
  - Agile Team Member
  - Architecture Owner

- **Secondary/optional roles:**
  - Domain Expert
  - Technical Expert
  - Independent Tester
  - Integrator
  - Specialist
Learning oriented

- Domain learning
  - Initial requirements envisioning
  - Incremental delivery of a potentially consumable solution
  - Active stakeholder participation throughout lifecycle

- Process improvement
  - Retrospectives at the end of an iteration
  - Tracking of improvements
  - Sharing of skills through non-solo development

- Technical learning
  - Architecture spikes
  - Proving the architecture with working code

- General strategies
  - Training
  - Education
  - Mentoring/coaching
  - Individuals are generalizing specialists, not just specialists
Disciplined Agile Delivery (DAD): A Hybrid Framework

DAD is a hybrid process framework that adopts best practices and philosophies from several methodologies.
The Disciplined Agile Delivery (DAD) process framework is a people-first, learning-oriented hybrid agile approach to IT solution delivery. It has a risk-value lifecycle, is goal-driven, scalable, and is enterprise aware.
Goals Throughout the DAD Lifecycle

<table>
<thead>
<tr>
<th>Goals for the Inception Phase</th>
<th>Goals for Construction Phase Iterations</th>
<th>Goals for the Transition Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Identify the vision for the project</td>
<td>- Produce a potentially consumable solution</td>
<td>- Ensure the solution is production ready</td>
</tr>
<tr>
<td>- Bring stakeholders to agreement around the vision</td>
<td>- Address changing stakeholder needs</td>
<td>- Ensure the stakeholders are prepared to receive the solution</td>
</tr>
<tr>
<td>- Align with enterprise direction</td>
<td>- Move closer to deployable release</td>
<td>- Deploy the solution into production</td>
</tr>
<tr>
<td>- Identify initial technical strategy, initial requirements,</td>
<td>- Maintain or improve upon existing levels of quality</td>
<td></td>
</tr>
<tr>
<td>and project plan</td>
<td>- Address highest risk(s)</td>
<td></td>
</tr>
<tr>
<td>- Set up the work environment</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Form initial team</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Secure funding</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Identify risks</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Ongoing Goals**

- Fulfill the project mission
- Grow team members skills
- Enhance existing infrastructure

- Improve team process and environment
- Leverage existing infrastructure
The coordinate-collaborate-conclude rhythm occurs at several scales on a disciplined agile delivery (DAD) project:

### Release rhythm

<table>
<thead>
<tr>
<th>Inception</th>
<th>Construction</th>
<th>Transition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coordinate</td>
<td>Collaborate</td>
<td>Conclude</td>
</tr>
</tbody>
</table>

### Iteration rhythm

<table>
<thead>
<tr>
<th>Iteration Planning</th>
<th>Development</th>
<th>Stabilize</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coordinate</td>
<td>Collaborate</td>
<td>Conclude</td>
</tr>
</tbody>
</table>

### Daily rhythm

<table>
<thead>
<tr>
<th>Coordination Meeting</th>
<th>Daily work</th>
<th>Stabilize</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coordinate</td>
<td>Collaborate</td>
<td>Conclude</td>
</tr>
</tbody>
</table>
The DAD Inception Phase

**Coordinate**
- Initiate team
- Schedule stakeholders for envisioning sessions

**Collaborate**
- Requirements envisioning
- Architecture envisioning
- Consider feasibility
- Align with enterprise strategy
- Build team
- Release planning (initial)
- Develop shared vision
- Setup environment

**Conclude**
- Light-weight milestone review
- Communicate vision to stakeholders

- Up to a few hours
- Ideally: Up to a few weeks
  - Average: 4 weeks
  - Worst case: Several months
- Up to a few hours
- Stakeholder consensus

Project Selected
The DAD Construction Phase

Coordinate
- Prove the architecture works via end-to-end working slice of the solution
  - Typical: 1 iteration
  - Worst case: Many iterations

Collaborate
- Incrementally produce a consumable solution
  - Several iterations
- Share project status with stakeholders
- Align with organizational goals
- Align with other project teams
- Improve individual and team performance

Conclude
- Determine sufficiency
- Harden the solution
  - Ideally: Several hours
  - Sufficient Functionality

Stakeholder Consensus
Proven Architecture
Typical Construction Iteration

- **Coordinate**
  - 2 hours for each week of the iteration length

- **Collaborate**
  - Typical: One to four weeks
  - Average: Two weeks
  - Worst case: Six weeks

- **Conclude**
  - One hour per week of iteration length
  - Potentially consumable solution

**Iteration planning**
- Visualize work
- Daily coordination meeting
- Refactoring
- Developer regression testing
- Model storming
- Continuous integration (CI)
- Sustainable pace
- Prioritized requirements
- Architecture spike
- Configuration management
- Burn-down chart
- Automated metrics

**Iteration modeling**

**“Standard” practices:**
- Test-driven development (TDD)
- Acceptance TDD (ATDD)
- Continuous deployment (CD)
- Look-ahead modeling
- Parallel independent testing
- Continuous documentation
- Non-solo development
- Look-ahead planning

**“Advanced” practices:**
- Iteration demo
- Retrospective
- Release planning (update)
- Consider sufficient functionality

5 October 2011
© Unified Process Mentors
Typical day during construction

- Daily coordination meeting
- Update task board
- Update iteration burndown

**Coordinate**
- Up to 15 minutes
- Start of day

**Collaborate**
- Address blocking issues
- Create tests
- Develop code
- Integrate
- Fix problems
- Model storm
- Deploy to test/demo environment
- Typical: 5-6 hours

**Conclude**
- Stabilize build
- Ideally: Not a concern
- End of day
The DAD Transition phase

- **Phase planning**
  - Transition planning
  - End-of-lifecycle testing and fixing
  - Pilot/beta the solution
  - Finalize documentation
  - Communicate deployment
  - Train/educate stakeholders

- **Coordinate**
  - Ideally: Nothing
  - Typical: One hour per week of collaborate time
  - Sufficient Functionality

- **Collaborate**
  - Ideally: Nothing
  - Average: 4 weeks
  - Worst case: Several months

- **Conclude**
  - Ideally: Less than an hour
  - Worst case: Several months
  - Production Ready
Enterprise awareness: Optimizing the whole

- Follow corporate conventions:
  - Standards and guidance for the architecture
  - Coding standards
  - Data guidelines
  - User interface (UI) guidelines
  - And many more

- Enhance the organizational ecosystem:
  - Reusing and leveraging the existing infrastructure is great
  - Enhancing and building out the infrastructure is better
  - Work closely with the enterprise architecture (EA) team

- Share learnings:
  - Personal and team improvement is great
  - Organization-level improvement is better
  - Your agile center of competency should harvest and share improvements

- Interact with other (potentially non-agile) teams:
  - Enterprise architecture
  - Data management
  - Governance
  - Quality assurance
  - Project management office (PMO)
Enterprise awareness: Governing agile teams

Agile teams provide:
- Significantly greater visibility to stakeholders regarding their actual status
- Many more opportunities for stakeholders to steer the project
- BUT… require stakeholders to be actively involved and accountable

Practices:
- Active stakeholder participation
- Potentially consumable solutions every iteration
- Risk-value lifecycle
- Explicit, light-weight milestone reviews
- Daily coordination meetings
- Iteration demos
- All-hands demos
- Follow enterprise development guidance
- Work closely with enterprise architects
- Automated metrics gathering
Disciplined Agile Delivery (DAD): The Foundation for Agility@Scale

**Team size**
- Under 10 developers
- 1000’s of developers

**Compliance requirement**
- Low risk
- Critical, audited

**Geographical distribution**
- Co-located
- Global

**Domain Complexity**
- Straight-forward
- Intricate, emerging

**Enterprise discipline**
- Project focus
- Enterprise focus

**Organization distribution (outsourcing, partnerships)**
- Collaborative
- Contractual

**Organizational complexity**
- Flexible
- Rigid

**Technical complexity**
- Homogenous
- Heterogeneous, legacy
Some agile whitepapers on IBM.com

• The Agile Scaling Model (ASM): Adapting Agile Methods for Complex Environments

• Scaling Agile: An Executive Guide

• Improving Software Economics: Top 10 Principles of Achieving Agility at Scale

• Enable the Agile Enterprise Through Incremental Adoption of Practices
Disciplined Agile Delivery (DAD) Resources

• **DAD Training**
  - IBM’s Advanced Disciplined Agile delivery workshop: 3 days (21 PDUs)
    • google “RP252 DAD”
    • (PMI approved, registered under provider number 1107)
  - The Disciplined Agile Delivery Experience: 3 days
    • [www.UPMentors.com/courses](http://www.UPMentors.com/courses)

• **DAD Services**
  - DAD Quick Start with IBM Rational Team Concert (RTC)
  - DAD w/ Atlassian Coaching, Consulting, Workshops
    • [www.AppFusion.com/Agile](http://www.AppFusion.com/Agile)

• **DAD Products**
  - The DAD process template for IBM Rational Team Concert (RTC)

• Or even better....
  - [www.DisciplinedAgileDelivery.com](http://www.DisciplinedAgileDelivery.com)
upcoming Disciplined Agile Delivery book

- IBM’s approach to Agile development
  - written by Mark Lines & Scott Ambler for IBM Press
  - Adds enterprise discipline to Scrum, Extreme Programming/XP and other agile techniques
  - available Q1 2012 on Amazon etc.
  - some chapters now available of Safari Roughcuts for review

- www.DisciplinedAgileDelivery.com
  - the DAD community website
  - *join the discussion today!!*
THANK YOU

Mark@UPMentors.com
Julian Holmes, Co-founder UPMentors, Julian@UPMentors.com (UK)
twitter.com/scottwambler, twitter.com/Mark_Lines
www.ibm.com/rational/agile