Disciplined Agile is an agnostic hybrid that leverages strategies from a variety of sources.

- SAFe®
- DevOps
- …and more
- PMIBok Guide®
- “Traditional”
- Agile Data
- Extreme Programming
- Unified Process
- Agile Modeling
- Scrum
- Kanban
- Lean
- DevOps
- Extreme Programming
- Unified Process
- Scrum
- SAFe®
- PMIBok Guide®
- “Traditional”
- Agile Data
- Agile Modeling
- …and more
Disciplined Agile (DA) is a process-decision toolkit that provides straightforward guidance to help people, teams, and organizations to streamline their processes in a context-sensitive manner.

DA provides a solid foundation for business agility by showing how the various activities such as Solution Delivery (software development), IT Operations, Enterprise Architecture, Portfolio Management, Security, Finance, Procurement and many others work together.

DA also describes what these activities should address, provides a range of options for doing so, and describes the trade-offs associated with each option.
Successful Agile Organizations Take Responsibility for Evolving their own Agility

• Success doesn’t come from adopting a prescriptive framework or methodology such as Scrum or SAFe, although it may be a good start!

• For true business agility, we need to “Choose our agile way of working (WoW)”, bespoke agile, optimizing for our unique situations

Choose Your Wow! is DA’s BOK for Disciplined Agile Delivery
The Seven Principles of Disciplined Agile

- Be Awesome
- Enterprise Awareness
- Optimize Flow
- Delight Customers
- Context Counts
- Choice is Good
The Discipline Agile (DA) Manifesto

We are uncovering better ways of working (WoW) by doing it and helping others to do it. Through this work we have come to value:

- **Individuals and interactions** over processes and tools
- **Consumable solutions** over comprehensive documentation
- **Stakeholder collaboration** over contract negotiation
- **Responding to feedback** over following a plan
- **Transparency** over (false) predictability

That is, while there is value in the items on the right, we value the items on the left more.
Disciplined Agile (DA) is described in four views.
Primary Roles on Disciplined Agile Delivery (DAD) Teams

Team Lead
Agile process expert, keeps team focused on achievement of goals, removes impediments

Product Owner
Owns the product vision, scope and priorities of the solution

Architecture Owner
Owns the architecture decisions and technical priorities, mitigates key technical risks

Team Member
Cross-functional team members that deliver the solution

Stakeholder
Includes the customer but also other stakeholders such as the sponsor, operations engineers, support staff, architecture, database groups, finance, and more
Your team interacts with others, and they need to become agile too!
Process blades encompass lots of practices.
Data Management Goal Diagram

Decision Points

- Improve Data Quality
- Evolve Data Assets
- Ensure Data Security
- Specify Data Structures
- Refactor Legacy Data Sources
- Govern Data

Database refactoring
- Continuous database integration
- Test-driven database development (TDD)
- Database regression testing
- Extract Transform Load (ETL)
- Meta data management (MDM)

Test data
- Reference data
- Master data
- Meta data

Data encryption
- Privacy access control
- Physical data security
- Accountability

Executable specifications
- Physical data model
- Enterprise conceptual model
- Enterprise data model
- Logical data model

Refactor database schema
- Announce refactoring
- Monitor deprecated schema
- Remove deprecated schema

Monitor and measure
- Develop data metrics
- Develop data guidance
- Data stewardship
Explore Usage

There are many ways to explore how people will work with our solution. Although there is significant focus within the Agile community on user stories and epics, these aren’t our only choices. Disciplined Agilists prefer to use the best technique for the situation they face, and as you can see in the table below there are several options available to us.

<table>
<thead>
<tr>
<th>Options</th>
<th>Trade-Offs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Epic</strong></td>
<td>Large stories that take a lot of effort, often multiple iterations, to complete. Epics are typically organized into a collection of smaller user stories (UI). Sometimes, Epics are referred to as Features or User Activities.</td>
</tr>
<tr>
<td><strong>Persona</strong></td>
<td>Detailed descriptions of fictional people who fill roles as stakeholders of the solution being developed.</td>
</tr>
</tbody>
</table>

**Trade-Offs**
- Useful for high-level program planning.
- Appropriate level of detail for low priority work since the details are likely not well understood yet and are likely to change anyway.
- Outcomes describe what stakeholders would like to achieve and why they would like to achieve that, but not how to do so.
- Provides teams flexibility in how to achieve the desired outcome.
- Useful to capture high-level stakeholder needs.
- Used as a technique to build empathy for users as real people, and to understand the optimal user experiences for each.
- Useful when we don’t have access to actual end users, or potential end users.
- Can be used as an excuse not to work with actual users.
- Puts use cases, and potentially usage scenarios and epics if we’re flexible, into context.
- Can promote requirements reuse via `<include>` and `<extend>` relationships.
- Can motivate unnecessary complexity via `<include>` and `<extend>` relationships.

**Usage scenario** describes the step-by-step interaction between a user/actor and the solution. Similar to acceptance criteria, although it tends to cross the equivalent of several stories. Also known as a use-case description.

**Mental Model Language (UML)** use case diagram. Diagrammatic notation for a textual use case (W, Object Primer).

**Story**
- Useful to flush out all the different ways that a solution can be used, often putting granular requirements such as stories or features into context.
- Danger of becoming a set of detailed requirements.
- Scenarios are typically less structured than...
The Process Goals of Disciplined Agile Delivery (Current)

**Inception**
- Form Team
- Align with Enterprise Direction
- Explore Scope
- Identify Architecture Strategy
- Plan the Release
- Develop Test Strategy
- Develop Common Vision
- Secure Funding

Get the team going in the right direction.

**Construction**
- Prove Architecture Early
- Address Changing Stakeholder Needs
- Produce a Potentially Consumable Solution
- Improve Quality
- Accelerate Value Delivery

Incrementally build a consumable solution.

**Transition**
- Ensure Production Readiness
- Deploy the Solution

Release the solution into production.

**Ongoing**
- Grow Team Members
- Coordinate Activities
- Address Risk
- Evolve Way of Working (WoW)
- Leverage and Enhance Existing Infrastructure
- Govern Delivery Team

Improve and work in an enterprise aware manner.
Teams should choose an appropriate lifecycle

- Agile
- Continuous Delivery: Agile
- Exploratory
- Lean
- Continuous Delivery: Lean
- Program
The Agile Lifecycle

- Inception: One or more short iterations, Stakeholder vision, Proven architecture
- Construction: Many short iterations producing a potentially consumable solution each iteration, Continued viability (several)
- Transition: One or more short iterations, Sufficient functionality, Production ready, Delighted stakeholders

Key steps:
- Daily Work
- Daily Coordination Meeting
- Initial Architectural Vision
- Iteration
- Tasks
- Consumable Solution
- Iteration wrap-up: Demo to stakeholders, Go-forward decision, Evolve our WoW
- Release solution into production
- Operate and support solution in production

Roadmaps & Guidance:
- Identify, prioritize, and select projects
- Envision the future
- Initial Vision and Funding
- Initial modeling, planning, and organization

Initial Requirements and Release Plan:
- Work Items
- Highest-Priority Work Items
The Continuous Delivery: Lean Lifecycle

- New Features
  - Business Value
  - Fixed Delivery Date
  - Expedite
  - Intangible Options

Envision the future
- Replenishment modeling session
- Roadmaps & Guidance
- Evolve WoW
- Process Experiments

Daily work
- Work items are pulled when capacity is available to address them
- Feedback
- Demo

Release solution into production
- Operate and support solution in production

Change Requests
- New Work
- Coordination Meeting
- Strategy

Construction
- Continuous stream of development
  - Sufficient functionality
  - Production ready
  - Delighted stakeholders
The Business Service Team Lifecycle

New Requests

Operational Support
Compliance
Expedite
Strategic Work Items

Replenishment session
Obtain feedback
Coordinate
Perform work
Pull work item
Process Experiments
Evolve WoW

Continuous Flow of Work

Finish
DA- FLEX: The Value Stream of the Effective Organization
DAD milestones – Lightweight and consistent

<table>
<thead>
<tr>
<th>Milestone</th>
<th>Fundamental Question Asked</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stakeholder vision</td>
<td>Do stakeholders agree with your strategy?</td>
</tr>
<tr>
<td>Proven architecture</td>
<td>Can you actually build this?</td>
</tr>
<tr>
<td>Continued viability</td>
<td>Does the effort still make sense?</td>
</tr>
<tr>
<td>Sufficient functionality</td>
<td>Has the team produced (at least) a minimum business increment (MBI)?</td>
</tr>
<tr>
<td>Production ready</td>
<td>Will the solution work in production?</td>
</tr>
<tr>
<td>Delighted stakeholders</td>
<td>Are stakeholders happy with the deployed solution?</td>
</tr>
</tbody>
</table>
Guided Continuous Improvement (GCI)

If we get better at this, we succeed more often and we improve faster.

We can do this if we have access to an experienced agile coach, but they're expensive and hard to find.

We can do this if we have access to a process knowledgebase, like the Disciplined Agile (DA) toolkit.

Some experiments fail. You learn something, but it’s still a failure.

Failing fast is fine, but succeeding early is better.

Identify Problem

Identify Potential Solution(s)

Try the Solution(s)

Assess Effectiveness

Adopt What Works

Abandon What Doesn’t Work

Share Learnings
Start where you are, do the best that you can in the situation that you face, and always strive to get better.
In Summary

- Disciplined Agile is a rich, comprehensive and well-organized toolkit of strategies to help your organization be more successful with Agile.
- DA brings a disciplined, agnostic, professional, enterprise approach to agile which is what our industry has been lacking in the past.
- Understanding what your options are, and which ones work in different contexts leads to better decisions.

“Better Decisions Lead to Better Outcomes”
To create the world’s most respected, comprehensive, credible, and worthwhile Agile Membership and Certification body in the world to enable organizations and community members to maximize their enterprise agility potential.
HOW TO FIND TRAINING & MORE INFORMATION ON DA MEMBERSHIP & CERTIFICATION

www.disciplinedagaileconsortium.org

Sign up for the free Newsletter!
THANK YOU

JONATHAN LEE, PMP, CSM, CDA
DA CHAMPION / PAST PRESIDENT
PMI CHICAGOLAND CHAPTER
DACHAMPION@PMICHICAGOLAND.ORG