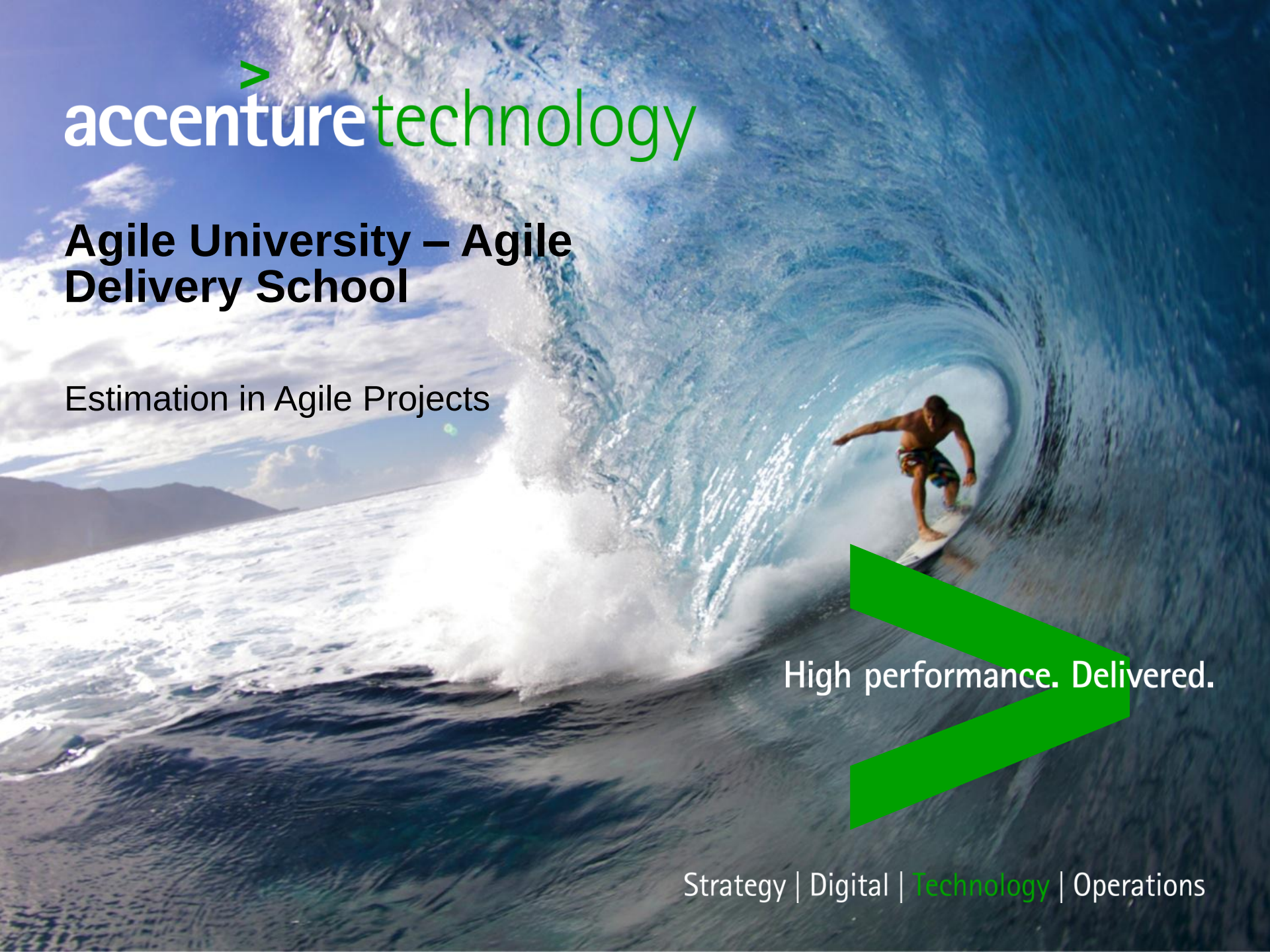

accenture technology
Agile University – Agile Delivery School
Estimation in Agile Projects
High performance. Delivered.
Strategy | Digital | Technology | Operations

Agenda

Estimation in Agile Projects

- Overview
- Release Planning - Estimate PBIs/User Stories
- Release Planning - Define Scope & Plan
- Sprint Planning - Identify Tasks & Estimate
- Mapping of Story Points to Effort
- Sprint Backlog - Updating ETC

ADM Agile Estimator

Estimation in Agile Projects - Overview

Step 1	**Project Initiation** - Product Owner defines the initial set of requirements as Product Backlog Items and prioritizes them.
Step 2.1	**Sprint 0** - Product Backlog Items (PBIs) will be refined in terms of clarification of requirements, decomposition of large Product Backlog Items (often called “epics”) into smaller ones (such as “user stories”).
Step 2.2	**Sprint 0** - Story Point estimation to Product Backlog Items to get an idea of the size of product backlog items.
Step 2.3	**Sprint 0** - Product Owner can re-prioritizes PBIs based on estimation.
Step 2.4	**Sprint 0** - Release Planning based on team’s velocity.
Step 3	**Spring Planning** - Team break the user stories into tasks and estimate the effort for each task (usually in person hours).

Release Planning - Estimate PBIs/User Stories

User Story	Priority	Story Points
User Story 1	Must Have	5 points
User Story 2	Should Have	8 points
User Story 3	Could Have	13 points
User Story 4	Should Have	5 points
User Story 5	Should Have	8 points
User Story 6	Could Have	3 points
User Story 7	Should Have	1 points
User Story 8	Must Have	5 points
User Story 9	Must Have	8 points
User Story 10	Could Have	13 points
User Story 11	Should Have	5 points
User Story 12	Must Have	8 points
User Story 13	Could Have	3 points
User Story 14	Must Have	13 points

- **Establish Story Point Baseline** – Identify user story from current product backlog or a different story which is completed earlier.
- **Confirm estimates** by comparing the story to multiple other stories.

Release Planning – Define Scope & Plan

Release Backlog	
User Story	**Story Points**
User Story 1	5 points
User Story 2	8 points
User Story 3	13 points
User Story 4	5 points
User Story 5	8 points
User Story 6	3 points
User Story 7	1 points
User Story 8	5 points
User Story 9	8 points
User Story 10	13 points
User Story 11	5 points
User Story 12	8 points
User Story 13	3 points
User Story 14	13 points
	98 Points

Release Plan			
IF Velocity = 25			
Sprint 1	**Sprint 2**	**Sprint 3**	**Sprint 4**
User Story 1	User Story 4	User Story 9	User Story 12
User Story 2	User Story 5	User Story 10	User Story 13
User Story 3	User Story 6	User Story 11	User Story 14
	User Story 7		
	User Story 8		

- **If team's velocity is known from a previous release** – if scope is fixed, # of iterations required to deliver the functionality = total size of release backlog / team's velocity.

  if deadline is fixed, then velocity multiplied by # of iterations to get an initial sense of how many features can be delivered.

- **If team's velocity is not known** – Rough estimate will be provided and the release plan will be less precise for the first few iterations, until a reliable velocity number can be derived.

Sprint Planning – Identify Tasks & Estimate

Example: Sprint Backlog

User Stories	Task	Owner	Estimate (hrs)
User Story 2 - Enable all users to place book in shopping cart	Design business logic	Sanjay	4
	Design user interface	Jing	2
	Implement back-end code	Tracy	6
	Implement front-end code	Joe	8
	Unit testing	Philip	4
	Regression testing	Philip	2
	Documentation	Tom	3
User Story 3 - Upgrade transaction processing module	Design user interface	Jing	4
	Set up shopping cart module	Tracy	2
	Implement back-end code	Tracy	2
	Implement front-end code	Joe	6

Team decides how much productive time it has available during the Sprint

Team breaks the user stories into tasks, estimates each task and assign owners.

Team decides how many User Stories items it can commit to complete during the Sprint

Mapping of Story Points to Effort

For example:

In the Sprint Backlog, for a particular sprint lets assume the following scenario	Number of user stories in that sprint = 5
	Assume that the sum of story points for those 5 user stories = 25 story points
	Absolute estimate using sprint backlog for those 5 user stories = 150 hours
	Available Capacity for that sprint = 120 hours
	Forecasted velocity = 25 story points

Sprint backlog is revisited to arrive at the realistic velocity	With the available capacity of 120 hours, team can achieve 4 user stories in that sprint.
	These 4 user stories sums up to 20 story points.
	Hence the realistic velocity based on the available capacity = 20 story points.

As the project matures (typically 5-6 sprints), the accuracy of story point estimation will increase.

Sprint Backlog – Updating ETC

				Day of Sprint					
Backlog Item	**Task**	**Owner**	**Initial Est.**	**1**	**2**	**3**	**4**	**5**	**6**
Enable all users to place book in shopping cart	Design business logic	Sanjay	4	2	0				
	Design user interface	Jing	2	2	1				
	Implement back-end code	Tracy	2	2	1				
	Implement front-end code	Joe	6	8	7				
	Unit testing	Philip	4	3	2				
	Regression testing	Philip	2	2	1				
	Complete documentation	Tom	8	6	2				
Upgrade transaction processing module	Design user interface	Jing	5	7	5				
	Set up shopping cart module	Tracy	6	4	2				
	Implement back-end code	Tracy	3	3	3				
	Implement front-end code	Joe	3	2	2				
	Total		**45**	**41**	**26**				

Agenda

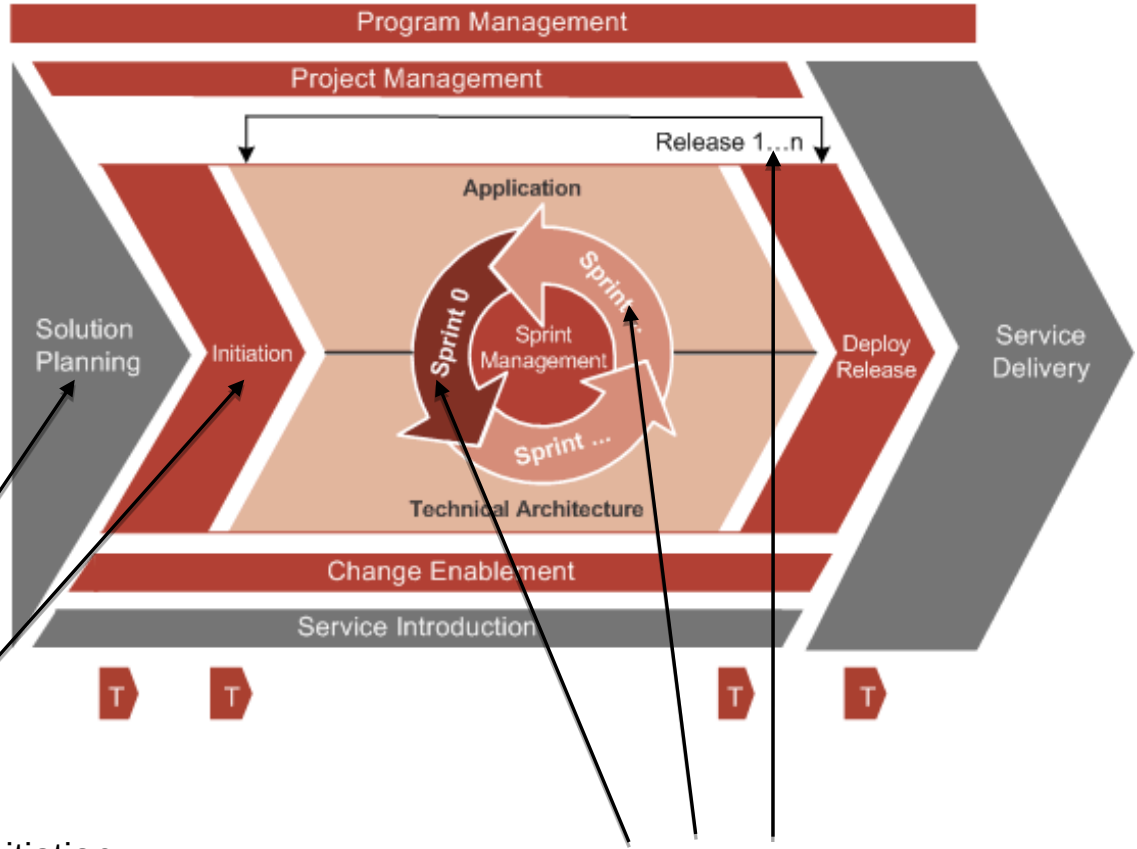
Estimation in Agile Projects

- Overview
- Release Planning – Estimating PBIs/User Stories
- Release Planning – Define Scope & Plan
- Sprint Planning – Identify Tasks & Estimate
- Sprint Backlog – Updating ETC

ADM Agile Estimator

ADM Agile Estimator

ADM Distributed Agile Estimator offers separate tools for **End-to-End** and **Sprint / Release** level estimation for projects aligned to ADM for Distributed Agile Methodology.



Solution Planning / Project Initiation

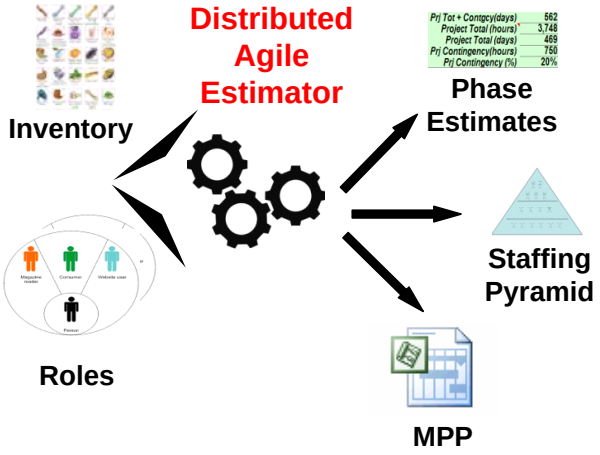
→ **ADM Distributed Agile Project Estimator**

Sprint / Release

→ **ADM Distributed Agile Sprint Estimator**

Distributed Agile Project Estimator

- End-to-End Estimator for Agile projects in Sales as well as Delivery stages.
- Can be used for collocated and distributed Custom Development Agile projects
- Includes Agile specific factors affecting the estimates, such as User Stories, Number of Sprints and Releases
- Aligned with Accenture's Distributed Agile Delivery Methodology
- Calibrated with Actual data from Distributed Agile projects executed at Accenture



Key Features

- Provides Estimates driven by Component Inventory, Multi-site and Agile factors
- Exports to Microsoft Project as a first cut plan
- Generates a Staffing pyramid based on the Roles supplied
- Easy-to-use and fine-tune the estimates
- Supported by comprehensive Guidelines for usage

Distributed Agile Release/Sprint Estimator

- Used at the beginning of a Release or Sprint
- Simple, easy to use, light-weight
- No macros, no add-ons for simplicity
- Provides estimates based on Technology specific inventory
- Platform/technologies supported: Java, .NET and Mobile
- Calibrated against 10 Agile projects at Accenture
- Planning Poker game can also be used to verify the estimates from Sprint Estimates.

Agile Packaged Project Estimation

- Current ADM Package estimators do not cover Agile specific factors
- ADM Distributed Agile estimator does not cover Package specific parameters
- The guidelines document complements the existing ADM estimators for Packaged development and provides guidance to fine-tune the estimates at phase and task level to include impacts of Agile while estimating

https://methodology.accenture.com/dist_agile/#meth.dist_agile/guidances/guidelines/Agile%20Packaged%20Estimation%20Guidelines_C23BA921.html