SWEN90016
Software Processes & Project Management

Project Scheduling

Marion Zalk
Department of Computing and Information Systems
The University of Melbourne
mzalk@unimelb.edu.au

2020 – Semester 1
Lecture 4

Copyright University of Melbourne 2018

Intended Learning Outcomes

1. Understand the role of a project schedule
2. Understand how to develop a project schedule
3. Understand how to use a project schedule to monitor and track project progress
4. Understand agile planning principles

Planning in Agile Development

- Takes a significantly different flavour from traditional approaches

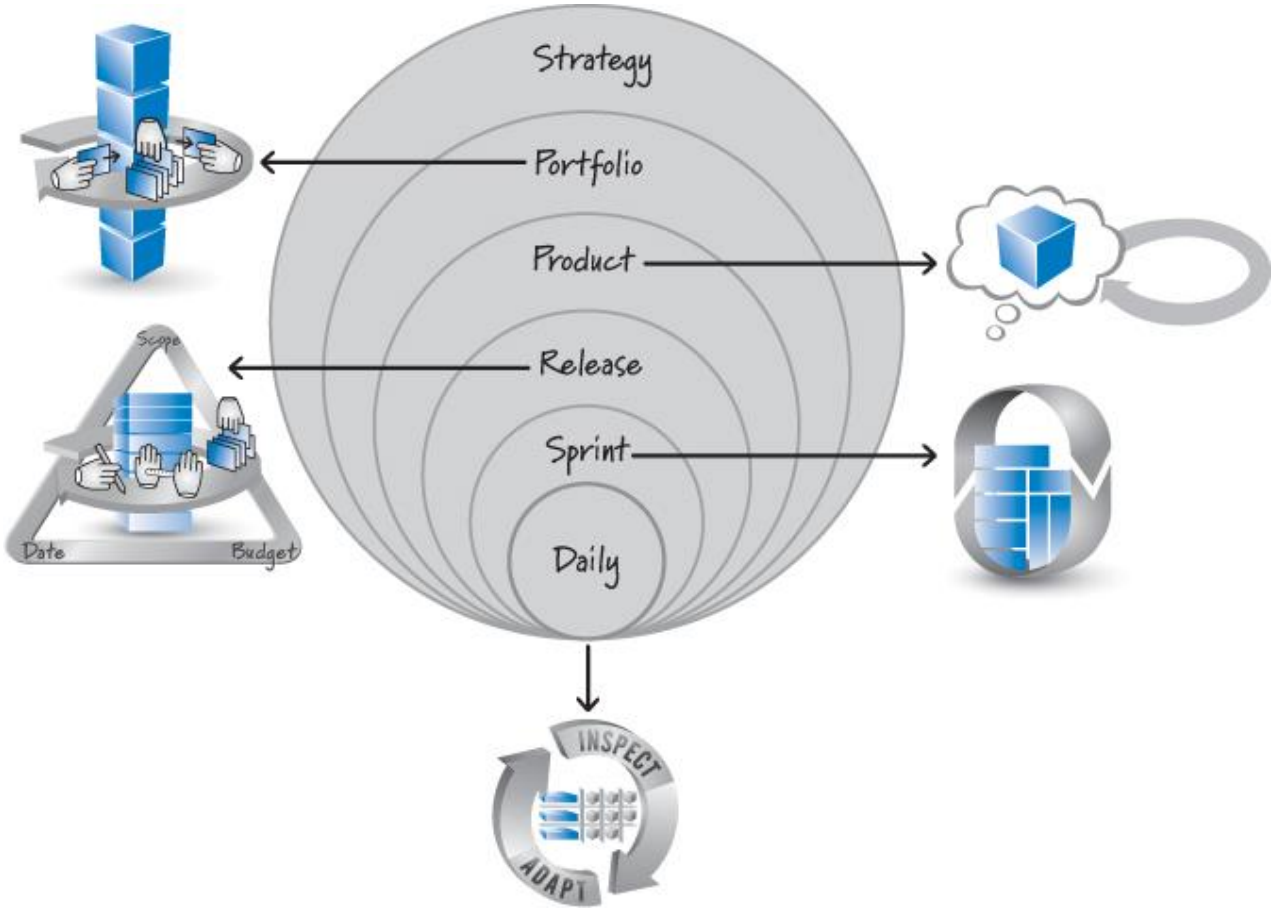
- Detailed planning is deferred until the start of the iteration
  - Designed to handle change
  - An iteration includes all phases (requirements, design and test)

- Planning is based on light weight lists
  - Gantt and PERT charts are considered less useful

Planning in Agile Development

- Plan short iterations
- Deliver working software
- Use “Just in time (JIT) planning” – next iteration
- Use the team

Planning in Scrum



Different levels of planning in Scrum

Planning in Scrum

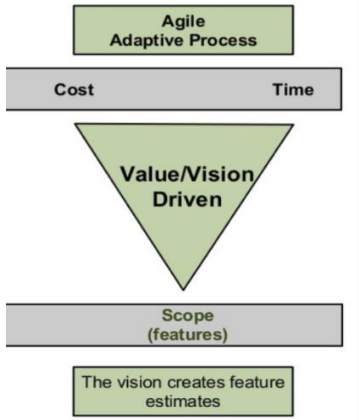
Level	Horizon	Who	Focus	Deliverables
Portfolio	Possibly a year of more	Stakeholders and product owners	Managing a portfolio of products	Portfolio backlog and collection of in-process products
Product (envisioning)	Up to many months or longer	Product owner, stakeholders	Visions and product evolution over time	Product vision, roadmap, and high-level features
Release	Three (or fewer) to nine months	Entire Scrum Team, Stakeholders	Continuously balance customer value and overall quality against the constraints of scope, schedule and budget	Release Plan
Sprint	Every iteration (one week to one month)	Entire Scrum Team	What features to deliver in the next Sprint	Sprint goals and sprint backlog
Daily	Every day	Scrum Master, development team	How to complete committed features	Inspection of current progress and adaptation

Release Planning

- Agile Planning
  - Recognizes that all three factors: scope, budget and time cannot be fixed in reality - not recommended
  - Can we fix scope and date and make the budget flexible?
    - Not really because increasing the budget, hence the resources will not always help to improve speed – not recommended
  - So what are our options?
    - Fix date and budget

      and have the scope flexible

      Fixed-Date release planning



    - Fix scope and have the date and budget flexible – *Fixed-Scope release planning*

Fixed-Date Release Planning

Determine the number of sprints N

$N = total\ duration/length\ of\ sprint$

↓

Groom the product backlog by estimating and prioritizing stories

↓

Measure team velocity range:

V_{min}, V_{max}

↓

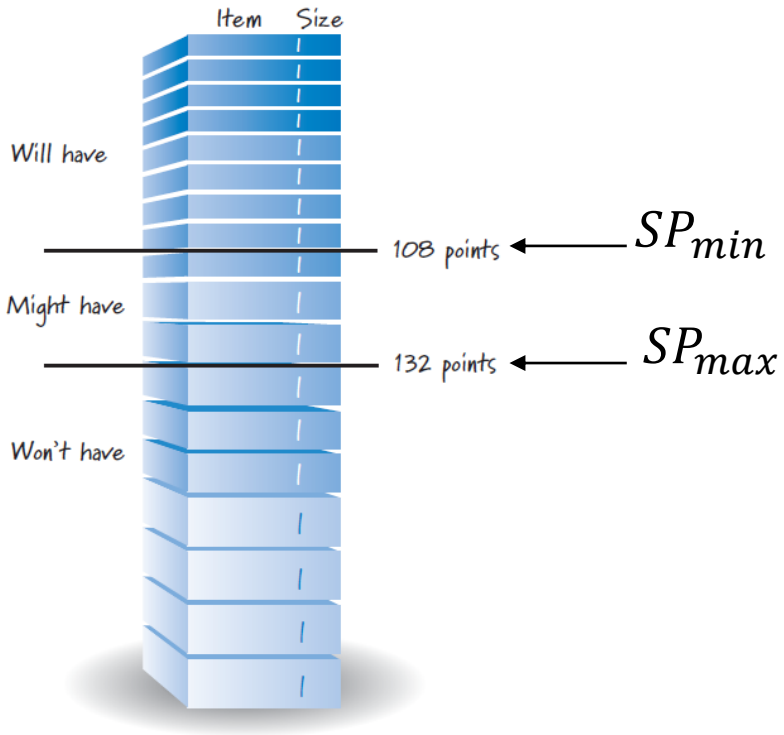
Compute minimum and maximum story points based on velocity

$SP_{min} = V_{min} \times N, SP_{max} = SP_{max} \times N$

↓

Draw lines through the Product Backlog to show the above

Fixed-Date: used when date is more important



Fixed-Scope Release Planning

Groom the product backlog by creating, estimating and prioritizing and identify the must-have stories

↓

Determine the total number of must-have story points (SP_{total})

↓

Measure team velocity range: V_{min}, V_{max}

↓

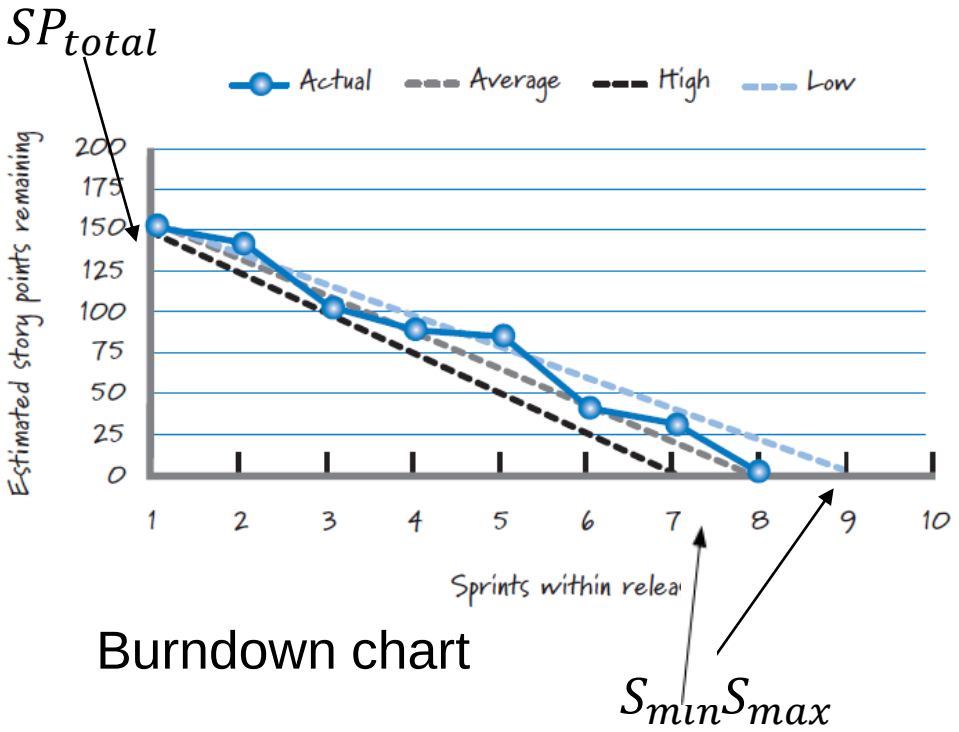
Compute minimum and maximum number of sprints

$$S_{min} = SP_{total}/V_{max},$$
$$S_{max} = SP_{total}/V_{min}$$

↓

Show on Burndown Chart

Fixed-Scope: used when scope is more important



Burndown chart

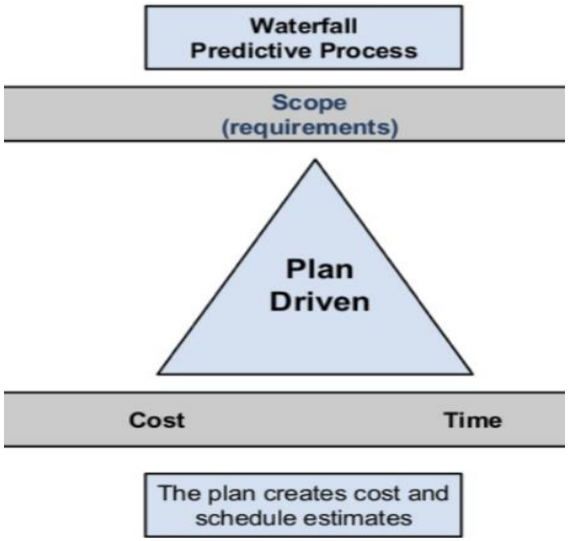
May require rounding up to be an integer

Intended Learning Outcomes

1. Understand the role of a project schedule
2. Understand how to develop a project schedule
3. Understand how to use a project schedule to monitor and track project progress
4. Understand agile planning principles

Release Planning

- Assumptions in Formal Planning:
  - Scope fixed – requirements are stable
  - Budget fixed – cost estimations are accurate
  - Schedule fixed - derived based on scope and budget



References

1. F. P. Brooks. The mythical man-month. In Essays on software engineering. Addison-Wesley, 1995.

2. R. S. Pressman. Software Engineering: A Practitioner's Approach. McGraw Hill, seventh edition, 2009.

3. Kenneth S. Rubin. Essential Scrum – A Practical Guide to the Most Popular Agile Process. Addison-Wesley, 2013.