6 : Scrum Planning

IT 4406 – Agile Software Developmet

Level II - Semester 4

Overview

- In this slide set SCRUM planning in the Agile Software Development is described in details.
- This lecture describes SCRUM planning principles, multi level planning along with the description of each and every level in that multi level planning.

Intended Learning Outcomes

- Understand the principals in Scrum Planning
- Describe how to apply scrum principles in planning.
- Understand principles behind Multi-level planning
- Discuss various Scrum planning activities and how they are interrelated with each other.
- Identify strategies for portfolio planning, grouped by scheduling, product inflow, and product outflow.

Intended Learning Outcomes

- Discuss how to determine whether or not more work should be invested in in-process products.
- Describe the envisioning approach used in Scrum principles.
- Discuss how release planning fits into the Scrum framework.
- Discuss how to perform release planning on both fixed-date and fixed-scope releases.

List of subtopics

6.1 Scrum Planning Principles

6.2 Multi level Planning

6.3 Portfolio Planning

6.4 Envisioning (Product Planning)

6.5 Release Planning (Longer-Term Planning)

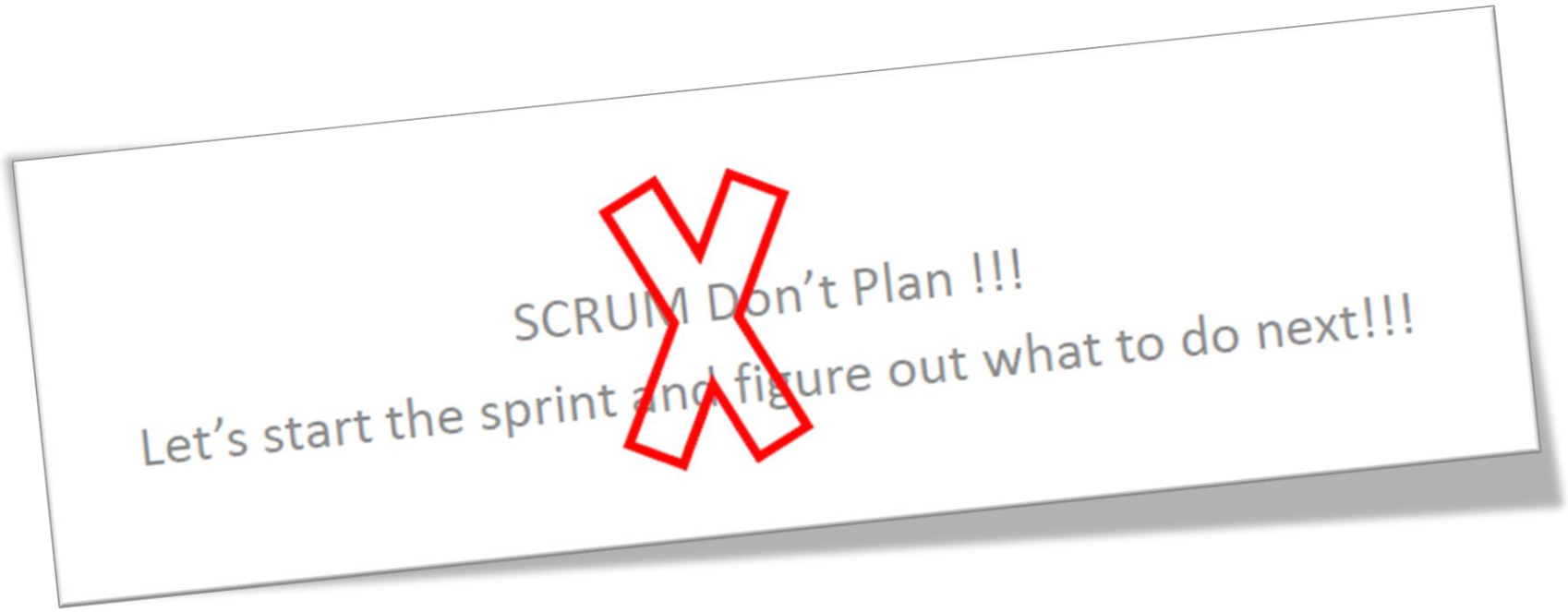
6.1 Scrum Planning Principles

- 6.1.1 Introduction
- 6.1.2 Don’t Assume We Can Get the Plans Right Up Front
- 6.1.3 Up-Front Planning Should Be Helpful without Being Excessive
- 6.1.4 Keep Planning Options Open Until the Last Responsible Moment
- 6.1.5 Focus More on Adapting and Preplanning Than on Conforming to a Plan

6.1 Scrum Planning Principles

- 6.1.6 Correctly Manage the Planning Inventory
- 6.1.7 Favor Smaller and More Frequent Releases
- 6.1.8 Plan to Learn Fast and Pivot When Necessary

SCRUM PLANNING



6.1.1 Introduction to Planning

Traditional Projects vs SCRUM Projects

- Traditional projects : Creates a detailed plan up front before development work begin.
  - ➢ get it right at the beginning so that rest can follow in an orderly fashion
- SCRUM : Does not believe it can be done in the very beginning
  - ➢ Can’t produce all of the planning artifacts up front
  - ➢ Up-front and just-in-time planning

SCRUM Planning Principles

- Keep important options open until the last responsible moment
  - Wait till you have more information to take critical decisions
- SCRUM believes up-front plan is helpful , as long as you are willing to read and adapt to the changes
- Frequent re planning
- Use validated learning to continuously produce better, more useful plans.

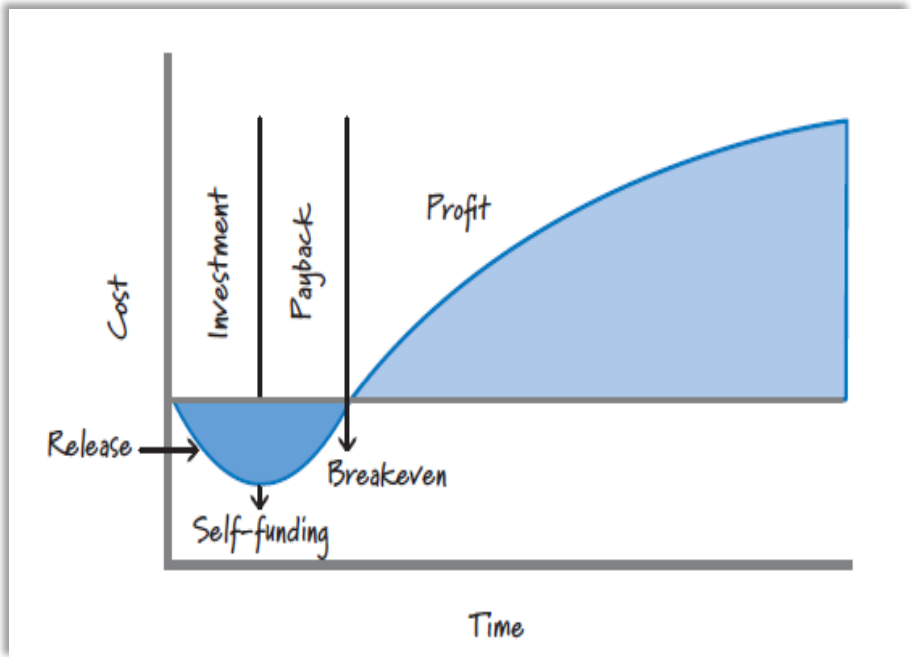
SCRUM Planning Principles

- Creating a large inventory of predictive, not-yet validated planning artifacts might be wasteful
- Three forms of waste
    1. wasted effort
    2. waste of having to update the plan
    3. wasted opportunity

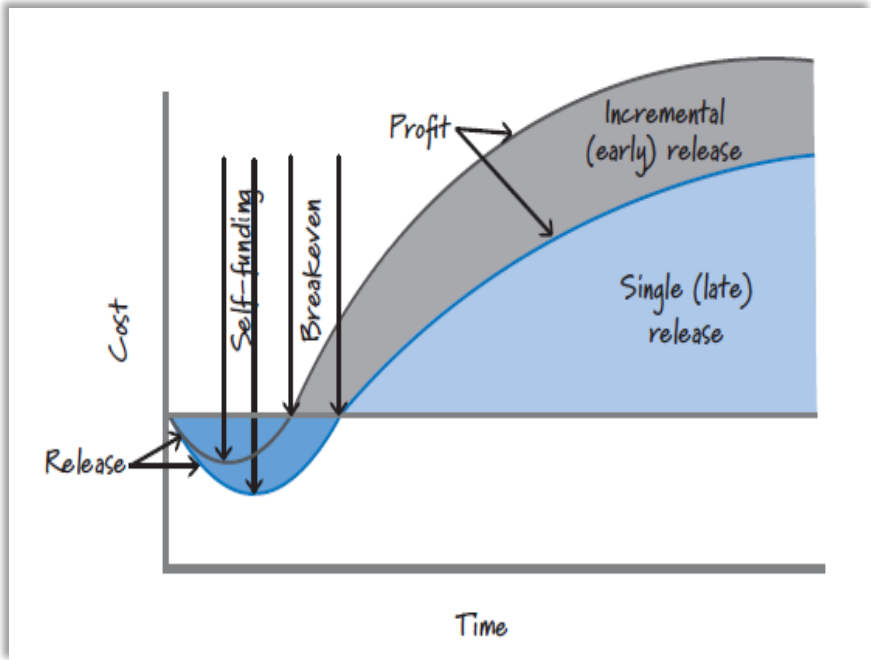
SCRUM Planning Principles

- Favors smaller, more frequent releases
  - ➢ provide faster feedback
  - ➢ improves ROI

Images are in the next two slides



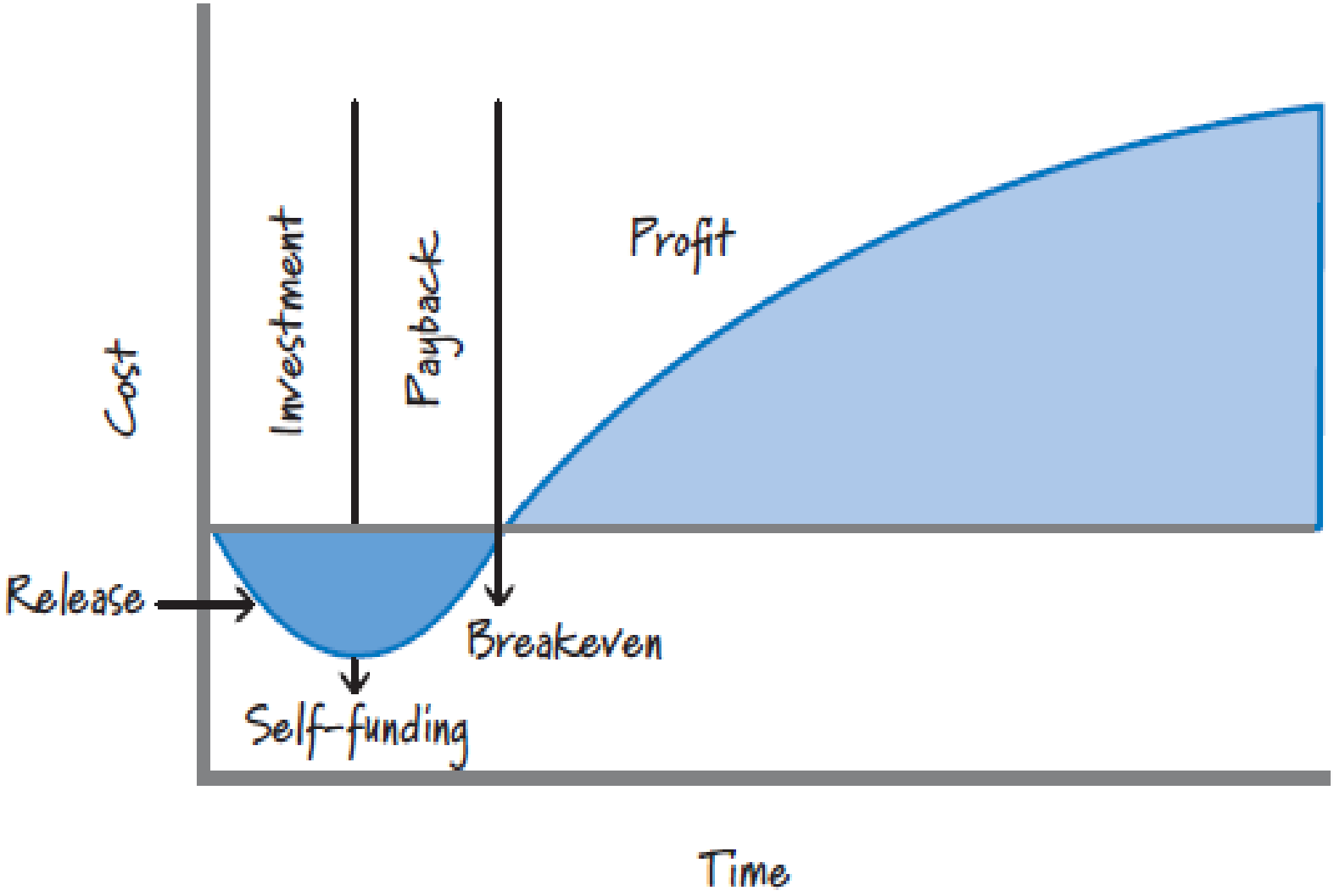
Single-release economics



Multi-release economics

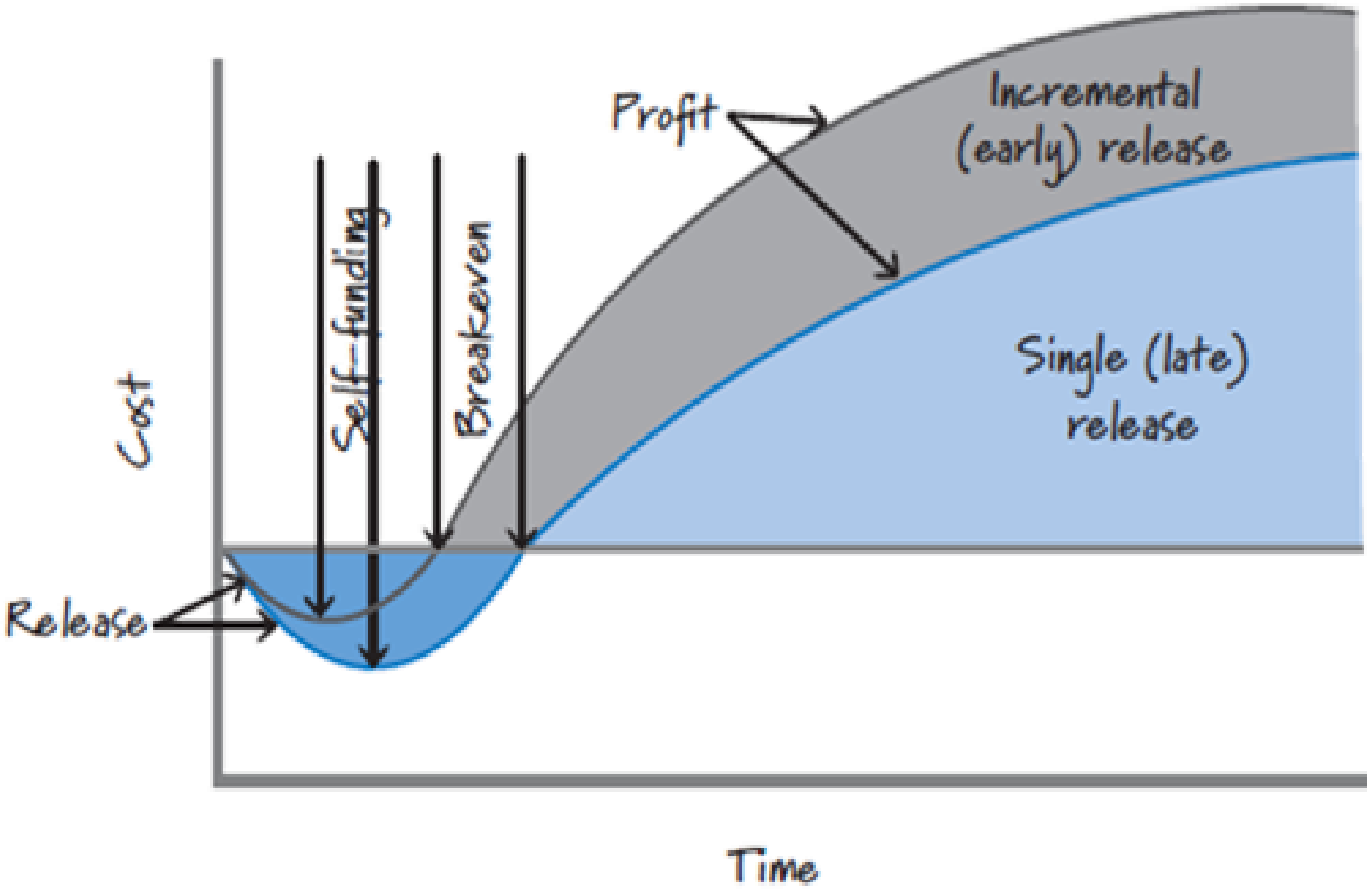
SCRUM Planning Principles

Single-release economics



SCRUM Planning Principles

Multi-release economics



SCRUM Planning Principles

- Plan to learn fast and pivot when necessary
  - ❑ Doing something
    - ➢ Learning fast
    - ➢ Changing directions if needed

6.2 Multi level Planning

- 6.2.1 Introduction
- 6.2.2 Portfolio Planning
- 6.2.3 Product Planning (Envisioning)
- 6.2.4 Release Planning
- 6.2.5 Sprint Planning
- 6.2.6 Daily Planning

Multi level Planning

- Plan at multiple levels of detail and at multiple times throughout development.
- Different levels of planning
  - Portfolio
  - Product
  - Release
  - Sprint
  - Daily

SCRUM formally define only these two

Portfolio Planning

- Determine which products to work on, in what order, and for how long.

Product Planning

- Product vision
  - provides a clear description of the areas in which the stakeholders get value.
- High-Level Product Backlog
- Product Roadmap

The outputs of product-level planning became inputs to portfolio planning

Release Planning

- Making scope, date, and budget trade-offs for incremental deliveries.
- To get an idea of what you can deliver by a fixed date -create and estimate a sufficient number of product backlog items
  - Draw a line through the product backlog
  - All of the items above the line are planned for the release
- Time dimension

Plan just in time and just enough

Sprint Planning

- Determine the specific product backlog items that the Scrum team will work on in the next sprint.

Daily Planning

- The most detailed level of planning
- Team members get together, and each person takes turns stating what he/she got done since the last daily scrum and the plan for today.

"Today I am going to work on the stored procedure task, and I should have that done by lunch. Whoever is going to work on the business logic task, please keep in mind that the business logic task is on the critical path for getting our work done this sprint and you should be ready to work on it right after lunch."

6.3 Portfolio Planning

- 6.3.1 Introduction
- 6.2.2 Scheduling Strategies

Portfolio Planning

- An activity for determining
  - which portfolio backlog items to work on
  - in which order
  - for how long
- Backlog item can be a product, a product increment, or a project

Portfolio planning deals with a collection of products and is therefore larger in scope and higher level than individual product-level planning

Portfolio Planning

- Timing
  - ❑ A never-ending activity
  - ❑ the output of product-level is an important input to portfolio planning.
    - ➢ determines whether to fund the product and how to sequence it into the portfolio backlog

Portfolio Planning

- Participants
  - portfolio planning focuses on both new products and in process products
  - internal stakeholders
  - the product owners of individual products
  - senior architects and technical leads

Portfolio Planning

- Process
  - ❑ Portfolio planning has two outputs
    - ➢ portfolio backlog
    - ➢ set of active products
  - ❑ To arrive at these outputs, participants engage in four categories of activities
    - ➢ Scheduling
    - ➢ managing inflows
    - ➢ managing outflows
    - ➢ managing in-process products

Portfolio Planning - Scheduling Strategies

- Optimize for lifecycle profits.
- Calculate the cost of delay.
- Estimate for accuracy, not precision.

Portfolio Planning - Inflow Strategies

- Apply the Economic Filter
- Balance the Arrival Rate with the Departure Rate
- Quickly Embrace Emergent Opportunities
- Plan for Smaller, More Frequent Releases

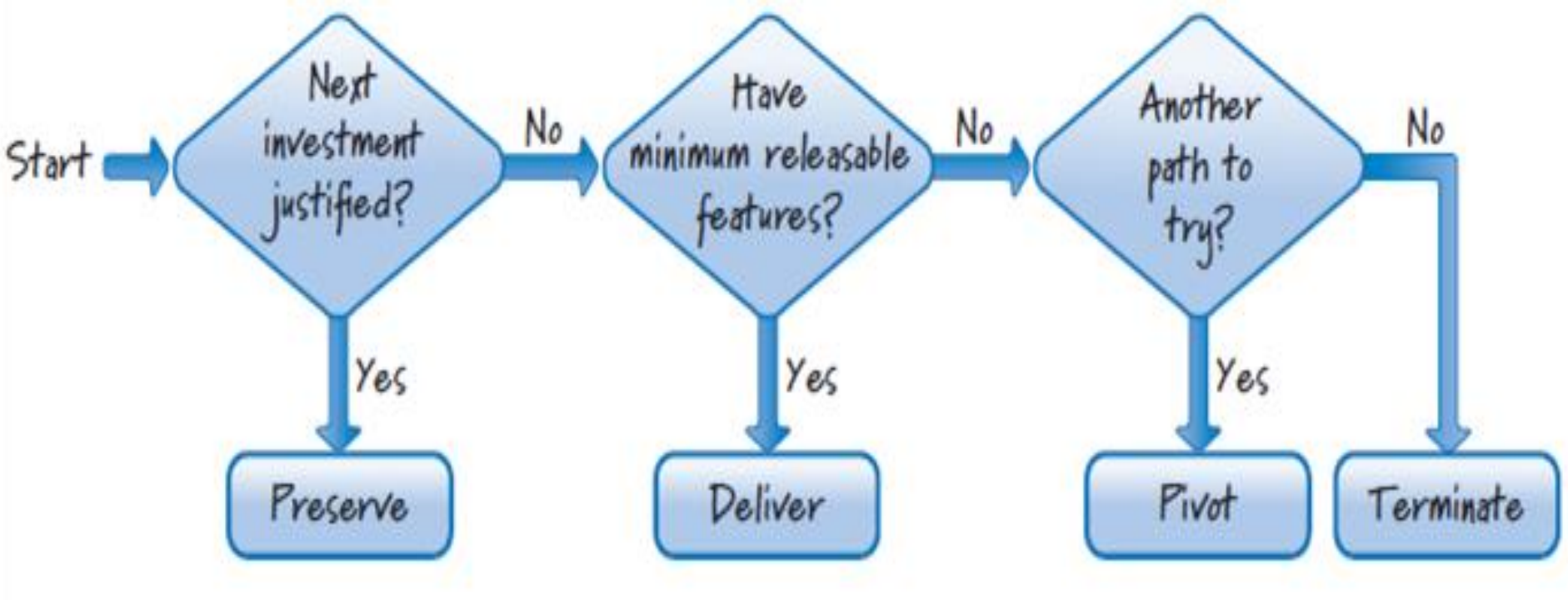
Portfolio Planning - Outflow Strategies

- Focus on idle work, not idle workers.
- Establish a WIP limit. (Work In Progress)
- Wait for a complete team.

Portfolio Planning - In-process Strategies

- Use Marginal Economics
  - Four options
    - Preserve - continue developing the product.
    - Deliver - stop working on the product and ship it.
    - Pivot - take what we have learned and change directions.
    - Terminate - stop working on the product and kill it.

Portfolio Planning - In-process Strategies



In-process product decision flow based on marginal economics

Ref. 1 - Page 284

6.4 Envisioning (Product Planning)

- 6.4.1 Introduction
- 6.4.2 SR4U Example
- 6.4.3 Visioning
- 6.4.4 High-Level Product Backlog Creation
- 6.4.5 Product Roadmap Definition
- 6.4.6 Other Activities
- 6.4.7 Economically Sensible Envisioning
- 6.4.8 Closing

Product Planning (Envisioning)

- Used to create the initial product backlog
  - ➢ a product vision is needed to generate an initial product backlog
- The goal is to work upon an idea, describing the essence of the potential product and creating a rough plan for how to approach its creation.
- Can decide whether to fund the next level of more detailed development

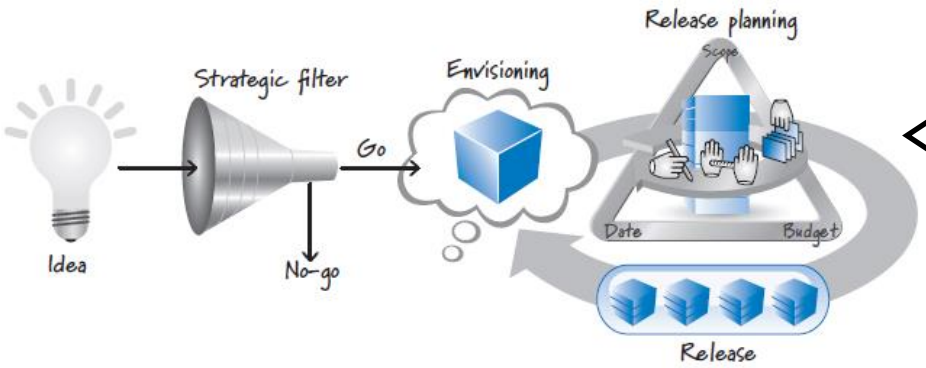
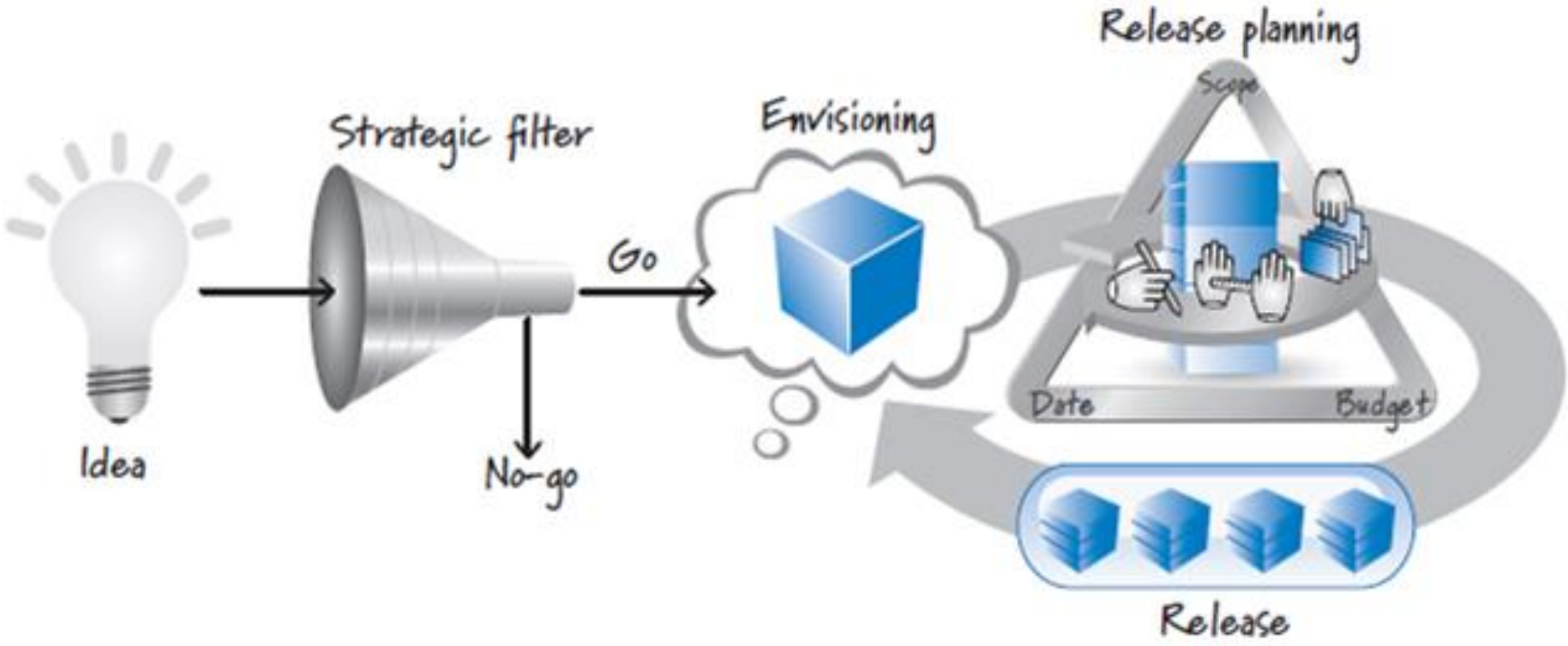


Image is in the other slide

Product Planning (Envisioning)



Product Planning (Envisioning)

- Participants
  - The product owner
  - internal stakeholders
  - Specialists in various areas

> Ideally, the Scrum Master and the development team should participate . But in reality, its not the case (Since companies waits till the initial envisioning is done to allocate the SCRUM team)

- Process
  - SR4U Example (Ref 1 - Page 290)

Ref 1 - Page 288

Product Planning (Envisioning)

- **Economically sensible envisioning**
  - ❑ Needs to be carried out in an economically sensible way.
  - ❑ It should be viewed as an investment.
  - ❑ Too little envisioning
    - ➢ unprepared to do the first customer-value-creation sprint
  - ❑ Too much envisioning
    - ➢ will create a large inventory of product-planning artifacts that may have to be reworked or discarded

Ref 1 - Page 288

Product Planning (Envisioning)

- **Guidelines for economically sensible envisioning**
  - Target a realistic confidence threshold
  - Focus on a short horizon
  - Act quickly
  - Pay for validated learning
  - Use incremental/provisional funding
  - Learn fast and pivot (aka Fail Fast)

Ref 1 - Page 288

6.5 Release Planning (Longer Term Planning)

- 6.5.1 Introduction
- 6.5.2 Release Constraints
- 6.5.3 Grooming the Product Backlog
- 6.5.4 Refine Minimum Releasable Features (MRFs)
- 6.5.5 Sprint Mapping (PBI Slotting)
- 6.5.6 Fixed-Date Release Planning
- 6.5.7 Fixed-Scope Release Planning
- 6.5.8 Calculating Cost
- 6.5.9 Communicating

Ref 1 - Page 307

Release Planning

- Helps answer
  - ❑ When will be done?
  - ❑ Which features can I get by the end of the year?
  - ❑ How much will this cost?
- Longer term planning
- Every organization must identify the proper sequence for releasing features
  - ❑ Different ways of releasing
    - ➢ release after multiple sprints
    - ➢ release every sprint
    - ➢ release every feature

Release planning involves collaboration between the stakeholders and the full Scrum team

Release Planning

- The inputs to release planning include outputs from product planning,
  - ❑ the product vision
  - ❑ high-level product backlog
  - ❑ product roadmap
- Activities
  - ❑ Confirm the release constraints of scope, date, and budget

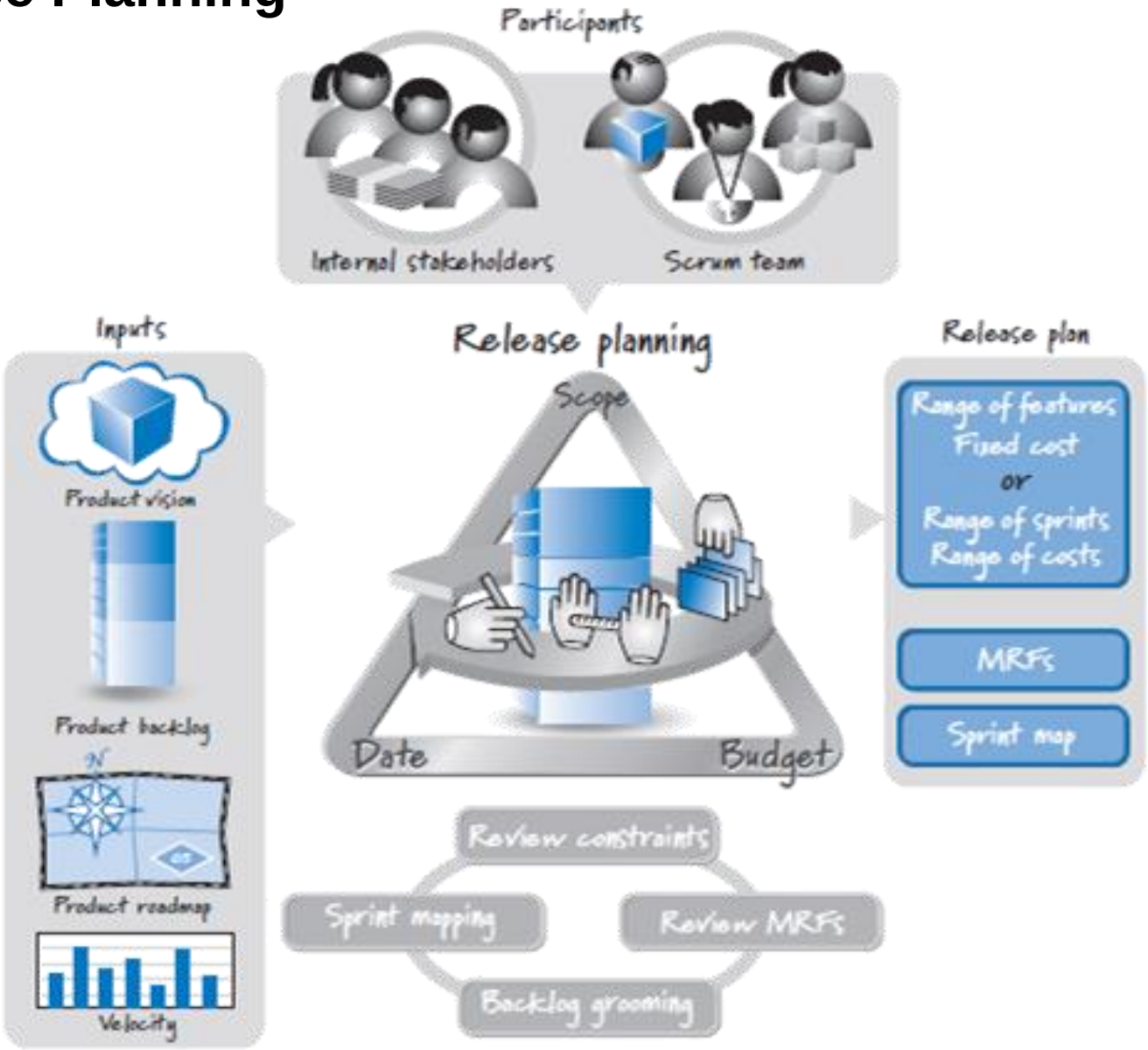
Release Planning

- Product backlog grooming
  - creating, estimating, and prioritizing product backlog items

Release Planning

- MRFs (Minimum Releasable Features)
- Always review the MRFs
- Some organizations also create a sprint map during this time

Release Planning



Release Planning Activity

Ref 1 - Page 310

Release Planning

- Release Constraints
  - ❑ The constraints of scope, date, and budget are important variables that affect the way project goals are achieved.
    - ➢ **Eg:** SR4U Release 1.0 had a fixed-date constraint
  - ❑ Development Constraint Combinations

Project Type	Scope	Date	Budget
Fixed everything (not recommended)	fixed	fixed	fixed
Fixed scope and date (not recommended)	fixed	fixed	flexible
Fixed scope	fixed	flexible	fixed(not really)
Fixed date	flexible	fixed	fixed

at least one of these variables should be flexible

Release Planning

- Grooming the Product Backlog (Contd…)
  - ❑ Once refined , the team would estimate them to determine rough cost.
  - ❑ Prioritize the estimated stories based on the release goal and constraints
- Sprint Mapping
  - ❑ In each sprint the team works on a set of product backlog items

Release Planning

- Early mapping of near-term product backlog items into sprints is helpful
  - Example...

Release Planning

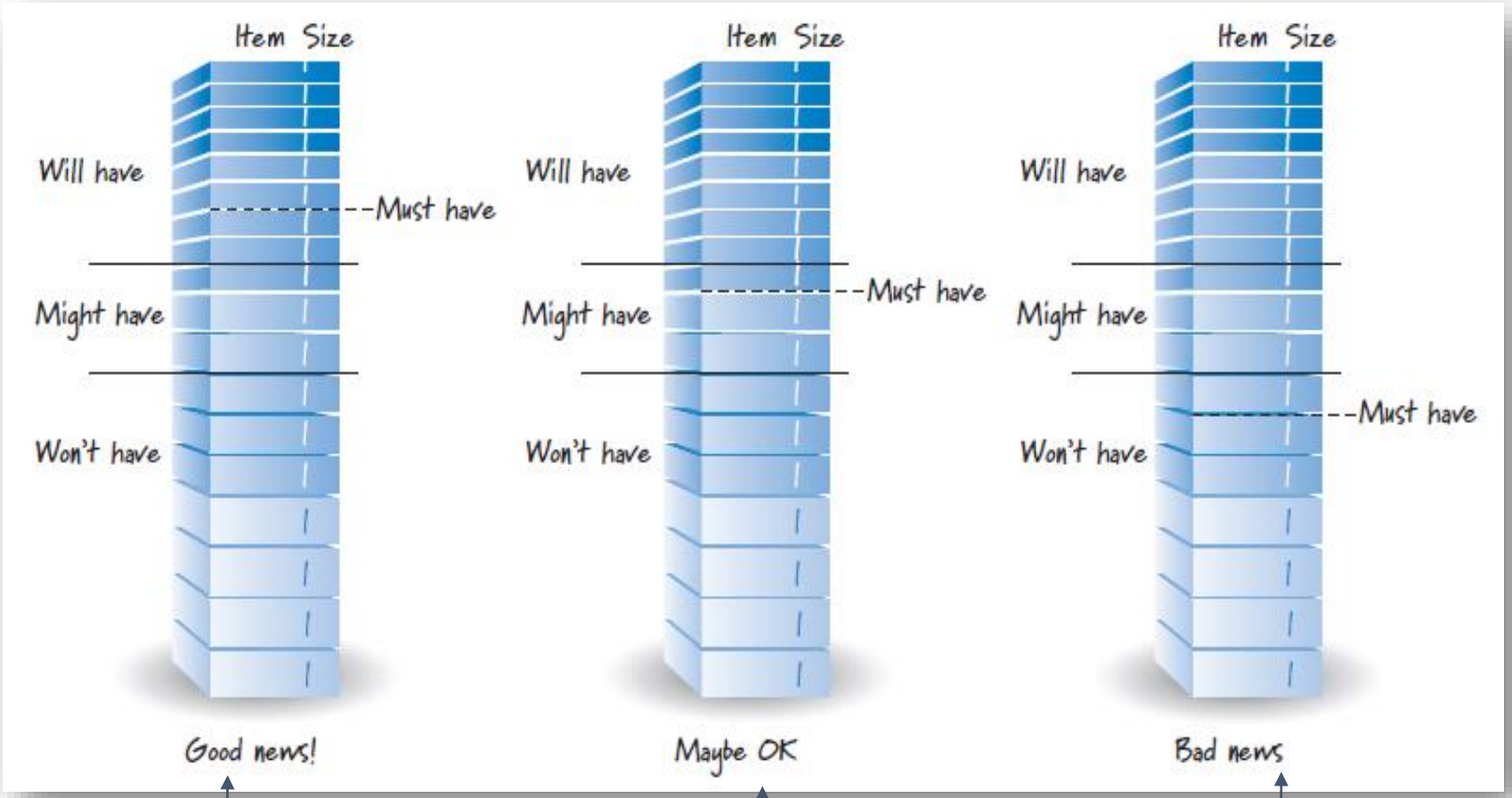
- Fixed-Date Release Planning
  - ❑ Many organizations prefer to use fixed-date releases.
    1. calculate the number of sprints needed to perform in this release (Assume six)
    2. determine how much work the team can get done in six sprints
    3. determine what features can be completed during the sprints
  - SR4U example

Release Planning

Steps for Performing Fixed-Date Release Planning

1. Determine how many sprints are in this release.
2. Groom the product backlog.
3. Measure or estimate the team's velocity.
4. Multiply the slower velocity by the number of sprints (will-have line).
5. Multiply the faster velocity by the number of sprints (might-have line).

Release Planning



We will have our must-have features

We will have most of our must-have features, but we might or might not have all of them

We won't have our must-have features

Ref 1 - Page 322

Release Planning

- Fixed-Scope Release Planning

what if your product scope is more important than the date??

what if you have a large set of must have features in your minimum releasable features set ??

Release Planning

calculate the number of sprints needed to perform in this release (Assume six)

1. determine how much work the team can get done in six sprints
2. determine what features can be completed during the sprints
    - SR4U example

Fixed-Scope Release Planning

Steps for performing Fixed-Scope Release Planning

1. Groom the product backlog
2. Determine the total size of the PBI(Product Backlog Item)s
3. Estimate the team's velocity as a range
4. Divide the total size of the PBIs by the faster velocity (round up)
5. Divide the total size of the PBIs by the slower velocity (round up)

Calculating Cost

- Identify the team.
- Determine the sprint length.
- Based on team composition and sprint length, determine the personnel costs of running a sprint.
- For a fixed-date release
  - multiply the number of sprints in the release by the cost per sprint.
- For a fixed-scope release
  - multiply both the high and low number of sprints by the cost per sprint.

Communicating

- Communicating progress on a fixed-scope release (Have an idea on the total scope to be achieved)
  - ❑ Fixed-Scope-Release Burndown Chart
    - ➢ shows the total amount of unfinished work

      Eg:

      - 150 story points at the start
      - At the end of each sprint, the chart has to be updated to show the total amount of work remaining
      - Sprint velocity
      - Can show projected outcomes

Communicating Progress on a Fixed-Scope Release : Fixed - Scope-Release Burndown Chart

- Three lines predicts, three release dates
- Highest velocity : 22 (will finish within 7 weeks)
- Lowest velocity : 18 (will take 9 weeks)

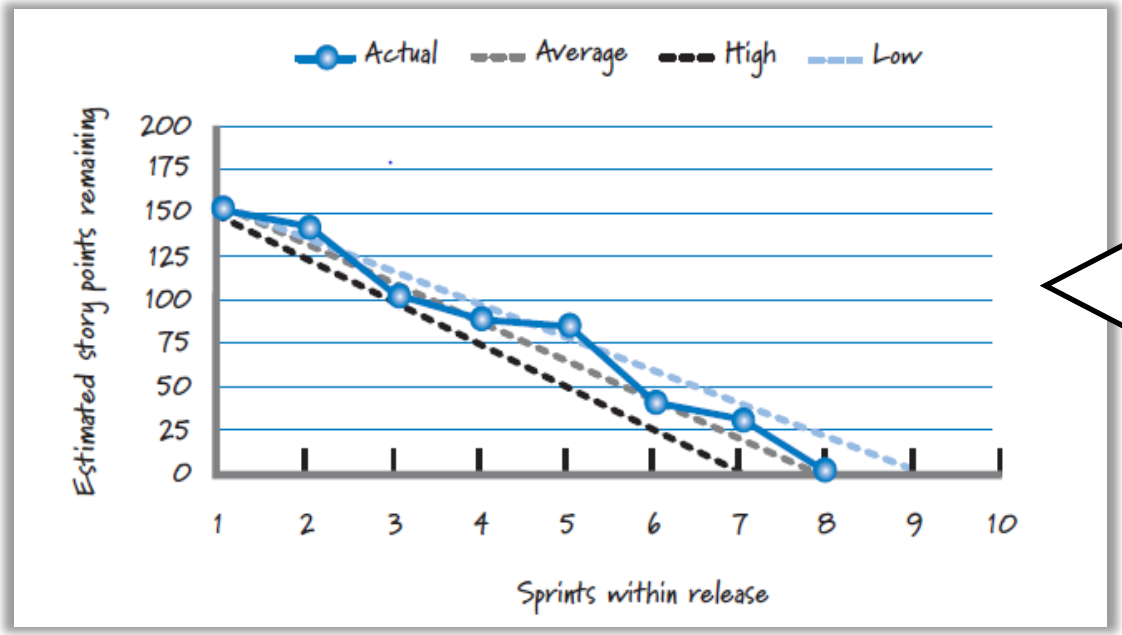
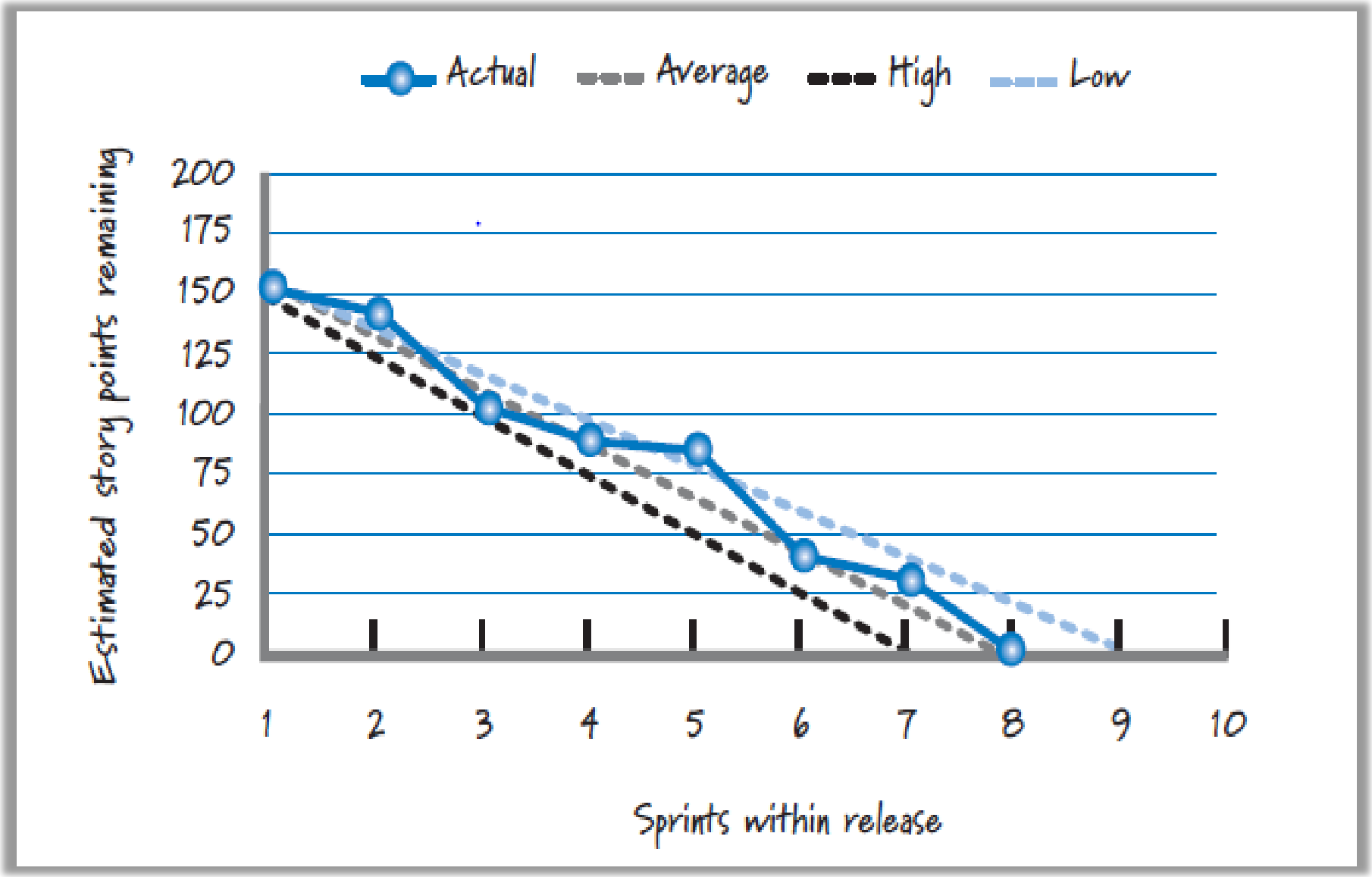


Image is in the next slide

Ref 1 : Page 327



Communicating Progress on a Fixed-Scope Release : Fixed-Scope-Release Burn Up Chart

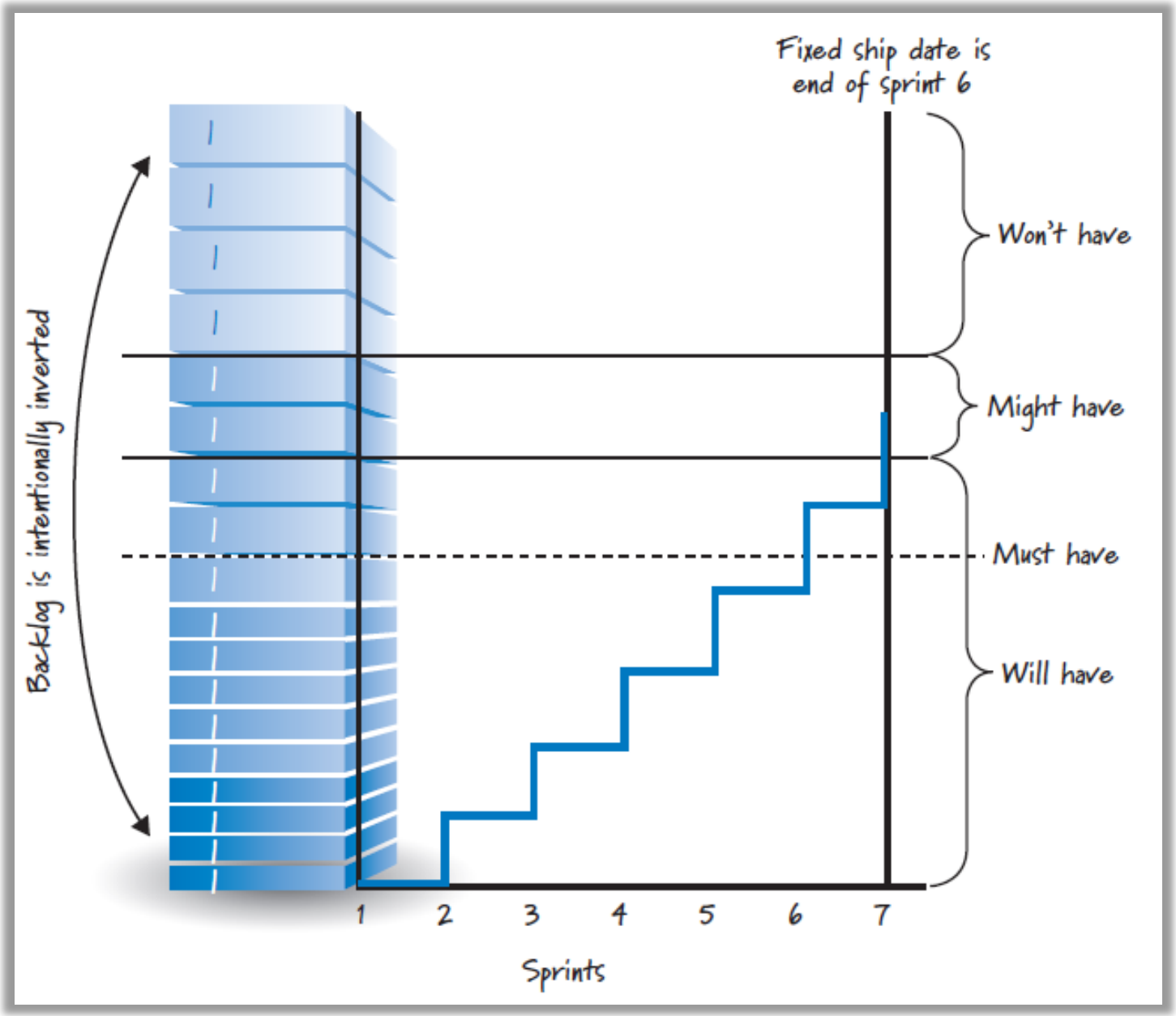
- Shows the total amount of work in a release as a goal and the progress of each sprint toward achieving that goal.

- Similarly, to the release burndown chart, this shows the same three predictive lines indicating the likely number of sprints to achieve the target.

Ref 1 : Page 327

Communicating Progress on a Fixed-Date Release

- Know the number of sprints in the release.
- Goal is to communicate the range of features we expect to complete and sprint by sprint progress.
- Traditional burndown and burn up charts are ineffective.
  - Why?
  - ...
- Need to update the chart created on fixed-date release plan at the end of each sprint.
- ***Can create a specialized burnup chart***



The product backlog is inverted

Highest-priority items at the bottom

Lower-priority items at the top

Summary

Scrum Planning Principles

Multi level Planning	Plan at multiple levels of detail and at multiple times throughout development.
Portfolio Planning	Determine which products to work on, in what order, and for how long.
Envisioning (Product Planning)	Goal is to work upon an idea, describing the essence of the potential product and creating a rough plan for how to approach its creation.
Release Planning (Longer-Term Planning)	Release planning involves collaboration between the stakeholders and the full Scrum team.

Reference

- Ref 1: [Addison-Wesley Signature Series (Cohn)] Kenneth S. Rubin - Essential Scrum_ A Practical Guide to the Most Popular Agile Process (2012, Addison-Wesley Professional)