Scrum: Estimation and Velocity

Lecturer: Adel Vahdati

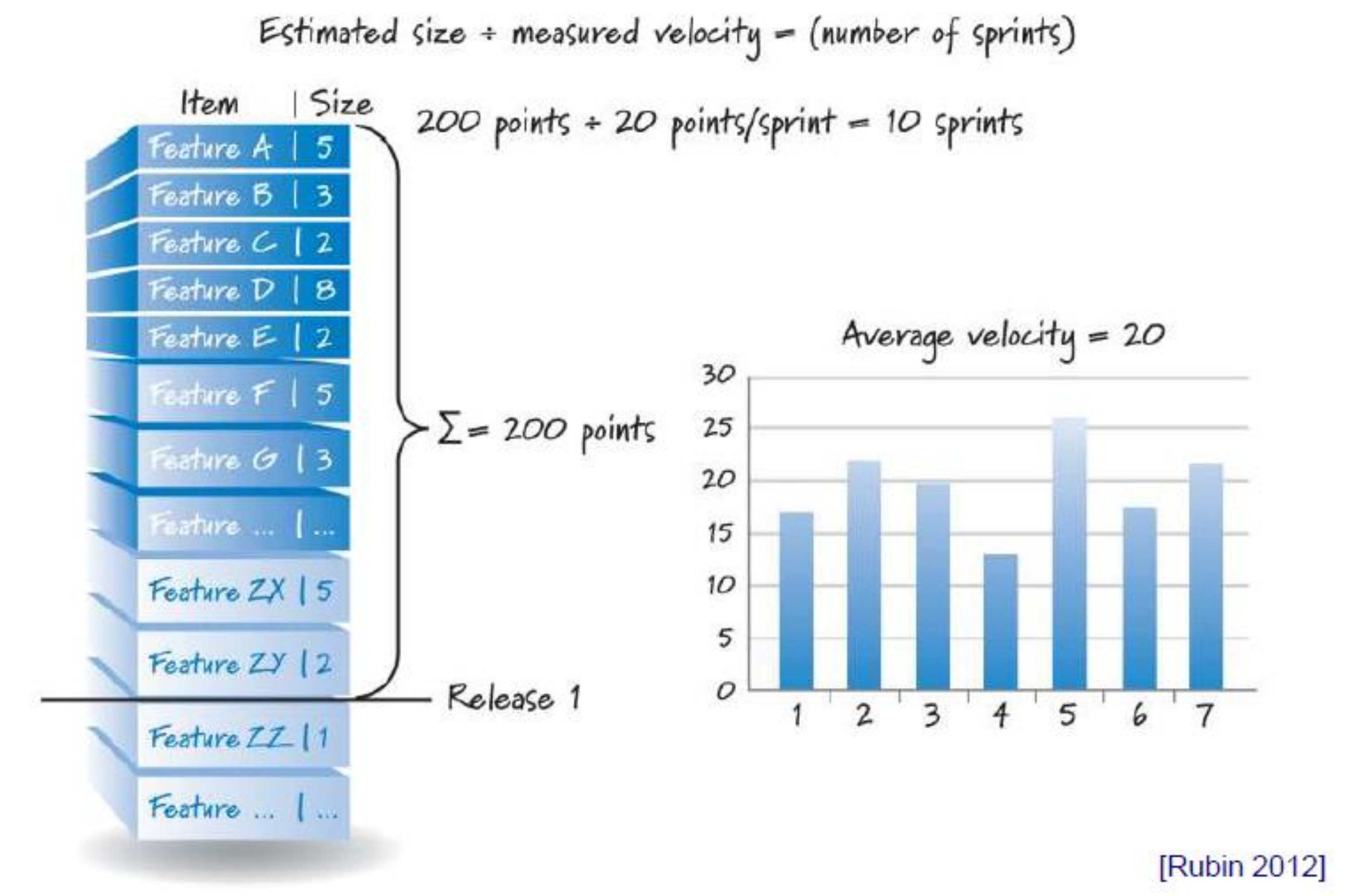
Estimation and Velocity

- When planning and managing the development of a product, we need to answer important questions such as:
  - "How many features will be completed?"
  - "When will we be done?"
  - "How much will this cost?"
- To answer these questions, we need to estimate the size of what we are building and measure the velocity at which we can get it done.
- With that information, we can derive the likely **product development duration** (and the corresponding cost) by dividing the **estimated size of a set of features** by the **team's velocity**.

Relationship Among Size, Velocity, and Duration

- Basic Question: How much time do we need to create the features in Release 1?
  - Gauge the **size** of Release 1 by adding the individual size estimates for the PBIs targeted for Release 1.
  - Estimate the team's **velocity**: How much work the team typically gets done each sprint.
    - At the end of each sprint, add the size estimates of the BPIs that were completed in the sprint; this sum is the team's velocity for that sprint.
    - Calculate an average velocity for the sprints that have been completed.
  - Now that we have estimated size and measured average velocity, calculate the **duration** by dividing the size by the velocity.

Relationship Among Size, Velocity, and Duration

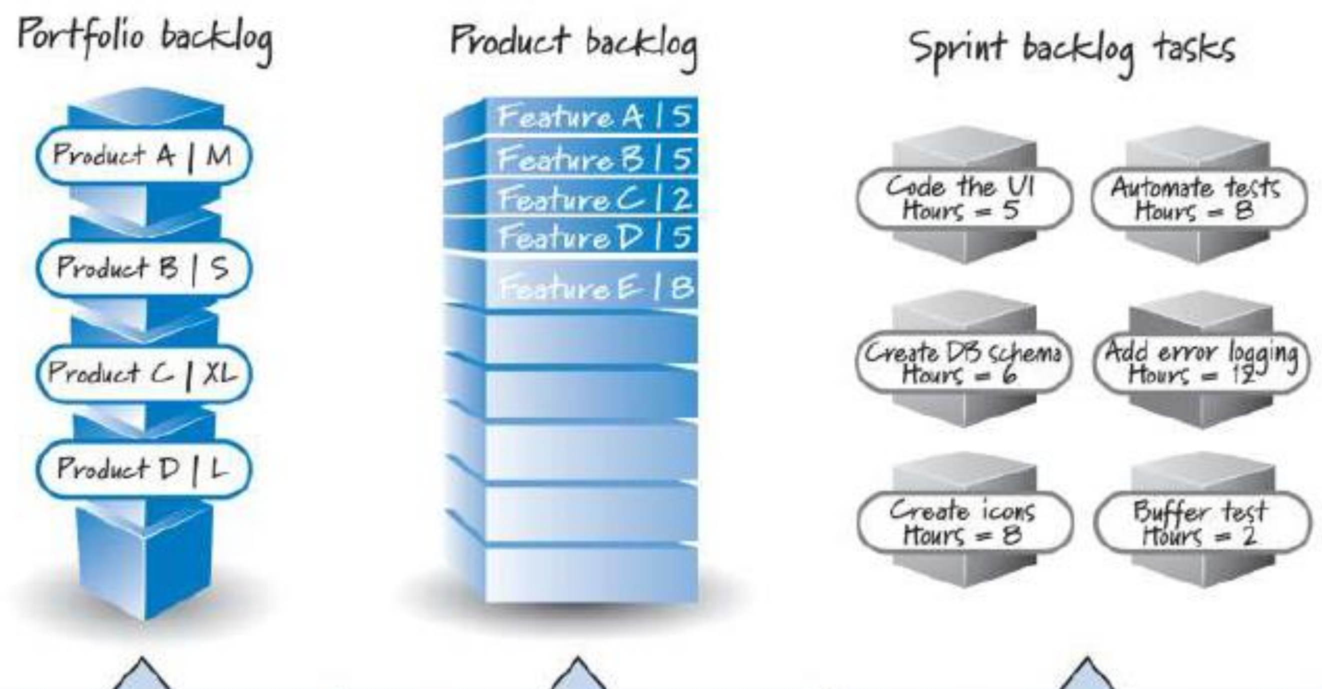


[Rubin 2012]

Estimation: What and When

- Throughout the development life of a product, we need to estimate at varying levels of granularity and, thus, will use different units to do so.
- Most organizations make estimates for planning purposes at three different levels of detail:
- **Portfolio backlog:** Contains a prioritized list of all of the products (or projects) that need to be built.
  - To estimate portfolio backlog items, rough, relative size estimates like T-shirt sizes are typically used (S, M, L, XL, and so on).
- **Product backlog:**
  - To estimate coarse-grained PBIs, T-shirt sizes are typically used.
  - For fine-grained PBIs which have risen in priority and been detailed, teams put relative numeric size estimates on them, using story points or ideal days.
- **Sprint backlog:**
  - Development tasks are typically sized in ideal hours (also referred to as effort-hours, man-hours, or person-hours).

Estimation: What and When



Item	Portfolio backlog	Product backlog	Sprint backlog tasks
Unit	T-shirt sizes	Story points / ideal days	Ideal hours / effort-hours
When	Portfolio planning	Product backlog grooming	Sprint planning

PBI Estimation Concepts

- **1. Estimate as a Team:** The development team that will do the work to design, build, and test the PBIs, will also do the estimation.
  - The product owner's role is to describe the PBIs and to answer clarifying questions that the team might ask.
  - The Scrum Master's role is to help coach and facilitate the estimation activity.
- **2. Estimates Are Not Commitments:** We do not want estimates to be artificially inflated due to external influences.
- **3. Accuracy versus Precision:** Our estimates should be accurate without being overly precise.
  - When estimating, there is a point of diminishing returns, beyond which we are wasting time and negatively affecting accuracy by considering lower-value data.
- **4. Relative Size Estimation:** We should estimate PBIs using relative sizes.
  - It has been observed that people are much better at relative size estimation than absolute size estimation.

PBI Estimation Units

- **Story points:** Measure the bigness or magnitude of a PBI.
  - Story points are influenced by factors such as complexity and physical size.
  - Story points combine factors like complexity and physical size into one relative size measure.
  - Because story points are ultimately used to calculate duration, they must reflect the effort associated with the story from the development team's perspective.
- **Ideal days:** Represent the number of effort-days or person-days needed to complete a PBI.
  - Ideal time is not the same thing as elapsed time; an important factor against using ideal time is this risk of misinterpretation.

Planning Poker

- Planning Poker is a consensus-based technique for estimating PBIs.
  - Experts slated to work on a PBI engage in an intense discussion to expose assumptions, acquire understanding, and size the PBI.
  - Planning Poker yields relative size estimates by accurately grouping or binning together items of similar size.
- To perform Planning Poker, the team must first decide which scale or sequence of numbers it will use for assigning estimates.
  - We favor a scale of sizes with more numbers at the small end of the range and fewer, widely spaced numbers at the large end of the range.
    - The most frequently used scale: 1, 2, 3, 5, 8, 13, 20, 40, and 100.
    - An alternative scale is based on powers of 2: 1, 2, 4, 8, 16, 32, ...
- We group or bin together like-size PBIs and assign them the same number on the scale.

Planning Poker: Participants

- The full Scrum team participates when performing Planning Poker.
  - The **product owner** presents, describes, and clarifies PBIs.
  - The **Scrum Master** coaches the team to help it better apply the game.
    - The Scrum Master is also constantly looking for people who seem to disagree and helping them engage.
  - The **development team** collaboratively generates the estimates.
- Each development team member is provided with a set of Planning Poker cards.

Planning Poker: Common Interpretation of Cards (1)

Card	Interpretation
0	Not shown in Figure 7.11 but included in some decks to indicate that the item is already completed or it is so small that it doesn't make sense to even give it a size number.
1/2	Used to size tiny items.
1, 2, 3	Used to size small items.
5, 8, 13	Used to size medium items. For many teams, an item of size 13 would be the largest they would schedule into a sprint. They would break any item larger than 13 into a set of smaller items.
20, 40	Used to size large items (for example, feature- or theme-level stories).
100	Either a very large feature or an epic.
∞ (infinity)	Used to indicate that the item is so large it doesn't even make sense to put a number on it.

Planning Poker: Common Interpretation of Cards (2)

Card	Interpretation
? (question mark)	Indicates that a team member doesn't understand the item and is asking the product owner to provide additional clarification. Some team members also use the question mark as a way of recusing themselves from the estimation of the current item—typically because the person is so far removed from the item he has no idea how to estimate it. Although it is acceptable not to estimate, it is unacceptable not to participate! So, just because someone doesn't feel comfortable offering up an estimate, that doesn't allow him to disengage from the conversation or responsibility of helping the team find a consensus estimate.
π (pi)	In this context, π doesn't mean 3.1415926! Instead, the pi card is used when a team member wants to say, "I'm tired and hungry and I want to get some pie!" Some Planning Poker decks use a coffee cup image instead of pi. In either case, this card emphasizes an important point. The team members can engage in an intense estimation discussion for only a limited period of time (perhaps an hour or two). At that point, they really do need a break or the enthusiasm for the discussion will turn into an effort to figure out how to quickly get the estimates done, regardless of their accuracy or the learning that takes place. If people are playing the pi card, the team needs to take a break.

Planning Poker: Playing Rules

- 1. The product owner selects a PBI to be estimated and reads it to the team.
- 2. Development team members discuss the item and ask clarifying questions from the product owner, who answers the questions.
- 3. Each estimator privately selects a card representing his estimate.
- 4. Once each estimator has made a private selection, all private estimates are simultaneously exposed to all estimators.
- 5. If everyone selects the same card, we have consensus, and that consensus number becomes the PBI estimate.
- 6. If the estimates are not the same, the team members engage in a focused discussion to expose assumptions and misunderstandings.
  - Typically we start by asking the high and low estimators to explain or justify their estimates.
- 7. After discussion, we return to step 3 and repeat until consensus is reached.

References

- Rubin, K.S., Essential Scrum: A Practical Guide to the Most Popular Agile Process, Addison-Wesley, 2012.
- Schwaber, K., Sutherland, J., The Scrum Guide, Published online at: http://www.scrumguides.org/, July 2013 (last visited on: 7 November 2014).
- Ramsin, Raman. "Home." Department of Computer Science and Engineering, Sharif University of Technology. Accessed February 15, 2025. https://sharif.edu/~ramsin/index.htm.