

D4 UNIT 2-A

ESTIMATION

Metrics and techniques



Basic concepts

Case 1

2

How much will you charge for the app I've just described?

1.000 euros.

Six months later...

I'm going to quote low



Basic concepts

Case 2

3

How much will you charge for the app I've just described?

I'm going to quote high

54.000 euros.

Ok, I'll think about it



Basic concepts

Case 3

4

OMG, the electricity bill is 127,40 euros

Have we got enough money for a new laptop that costs 1.200 euros?

The following day...

Yes, we've got 2.000 euros left.



Basic concepts

Software vs other industries

5

A company needs to estimate the cost and the time needed to sew 2000 jeans:



Productivity of each worker	5 jeans/hour		
Direct cost of a worker	12 €/hour	Days (2 workers)	25
Indirect cost of a worker	8 €/hour	Cost	9.500 euros
Transportation and management	1.500 €		

What if only a worker is available?

What if they were 3?

What else should be considered?

Basic concepts

Challenges in software estimation

6

- How is the required effort estimated?
- How is the productivity of a software architect, a software designer or a programmer measured?
- Wide deviations in the productivity of a worker.
- Same functionality but different attributes imply different costs (encryption, modularity...).
- Software requirements volatility.
- Constant innovation.

Basic concepts

Estimation precision and accuracy

7

- Accuracy is the closeness of agreement between a measured value and a true or accepted value. Measurement error is the amount of inaccuracy.
- Precision is a measure of how well a result can be determined (without reference to a theoretical or true value).
- The uncertainty estimate associated with a measurement should account for both the accuracy and precision of the measurement.

https://www.webassign.net/question_assets/unccolphysmechl1/measurements/manual.html

Basic concepts

Estimation precision and accuracy

8

Tom Cargill's Ninety-ninety rule:

"The first 90% of the code accounts for the first 90% of the development time.

The remaining 10% of the code accounts for the other 90% of the development time"

Are our estimates usually optimistic or pessimistic?

Do we assess the most complex parts in an appropriate way?

Basic concepts

Estimation precision and accuracy

9



How many ml does each one contain?

What is the ratio between both?

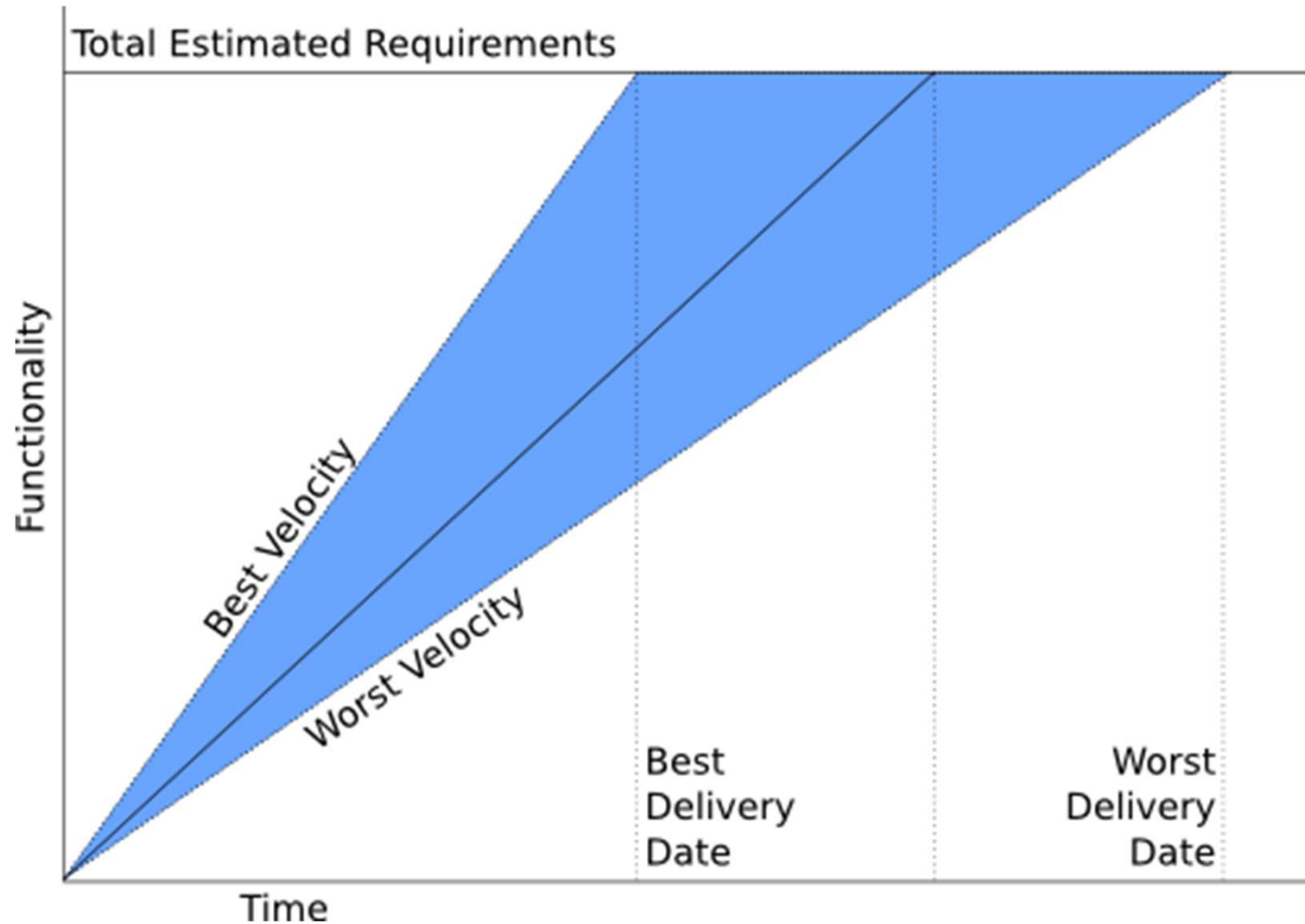
BlackPepper

<https://www.blackpepper.co.uk>

Basic concepts

Estimation precision and accuracy

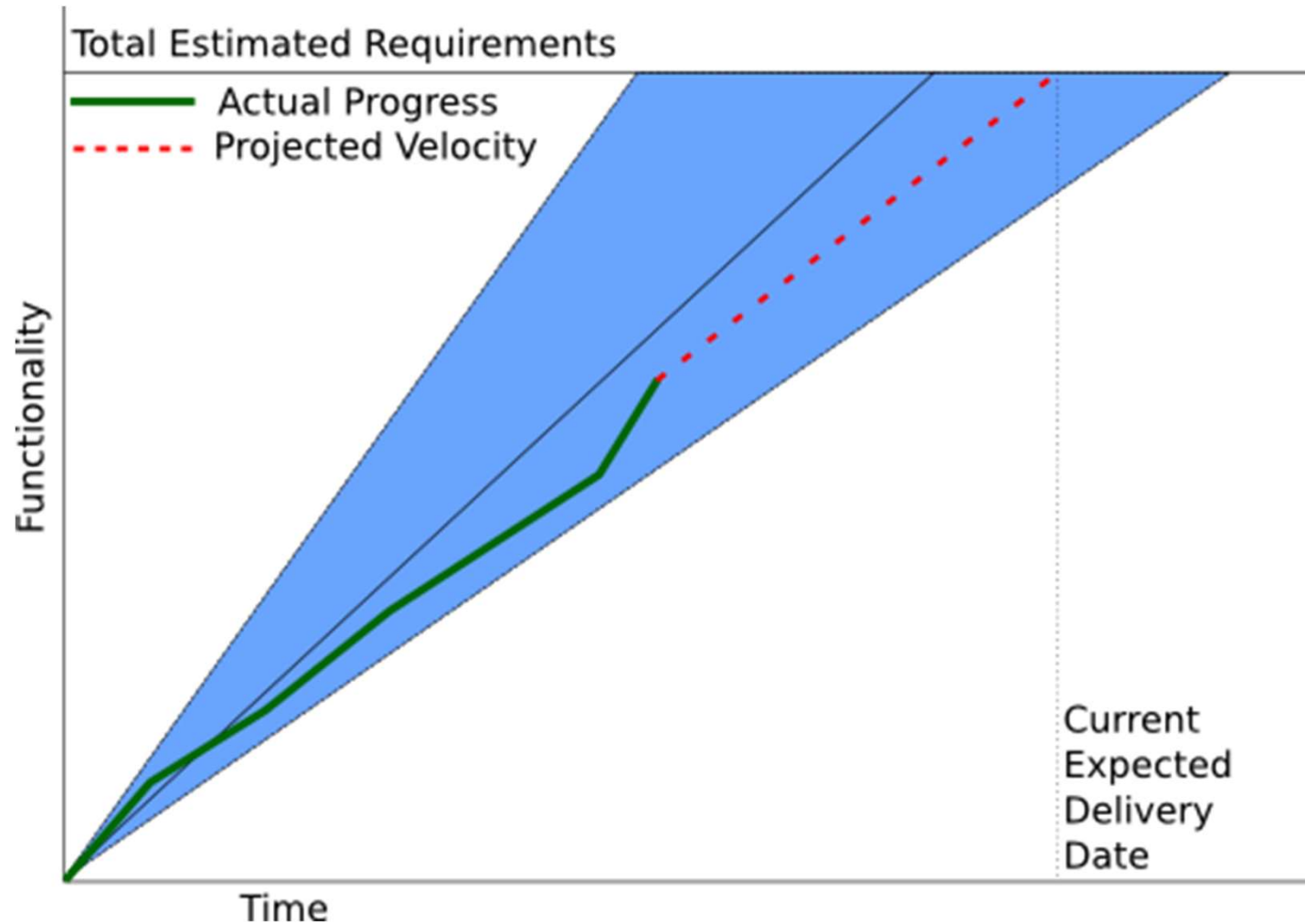
10



Basic concepts

Estimation precision and accuracy

11



Estimation phases

Estimations to perform

12

- What amount of effort will be required?
- How many people will be involved?
- What software and hardware resources will be necessary?
- How long will it take?
- How much will it cost?
- What risks will be implied?

Estimation phases

Main sections

13



Estimation phases

Main sections

14

1. **Size estimation:** How much do I have to develop? How complex is it?
2. **Effort estimation:** How long will it take? How many people do I need?
3. **Cost estimation:** Identification of third-party resources, cost of the resources, budget and quotation.

Estimation phases

Productivity estimation

15

- It is necessary to estimate productivity to estimate the project.
- It is based on the measurement of certain software attributes.

Estimation phases

Aspects that affect duration

16

- Size of the project.
- Complexity of the project.
- Experience in the application domain.
- Process quality.
- Technological support.
- Working atmosphere.
- Uncertainty.

Even though, individual differences are among the most important factors.

Metrics

Things to measure

17

*Measure what can be measured, and make
measurable what cannot be measured.*

Galileo Galilei

Metrics

Definitions

18



Entity

Object to be characterized

Attribute

Measurable characteristic of an entity

Measurement

Process to assign values or symbols

Measure (measured quantity value)

Value or symbol assigned in the measurement

Metrics

Scales

19

Nominal

Java, C++, Python, Ruby

Dichotomic

Yes/no, true/false

Ordinal

Nothing, Few, Intermediate, Much, All

Interval

0-10, 11-20, 21-30, >30

Absolute

0,1,2,3,4,5

Ratio

0,1,2,3,4,5 with the possibility to compare proportions

Metrics

Things to measure in SW Engineering

20

- Products

Size, complexity, reusability, coupling, test cases, quality, reliability, maintenance...

- Processes

Time, effort, cost...

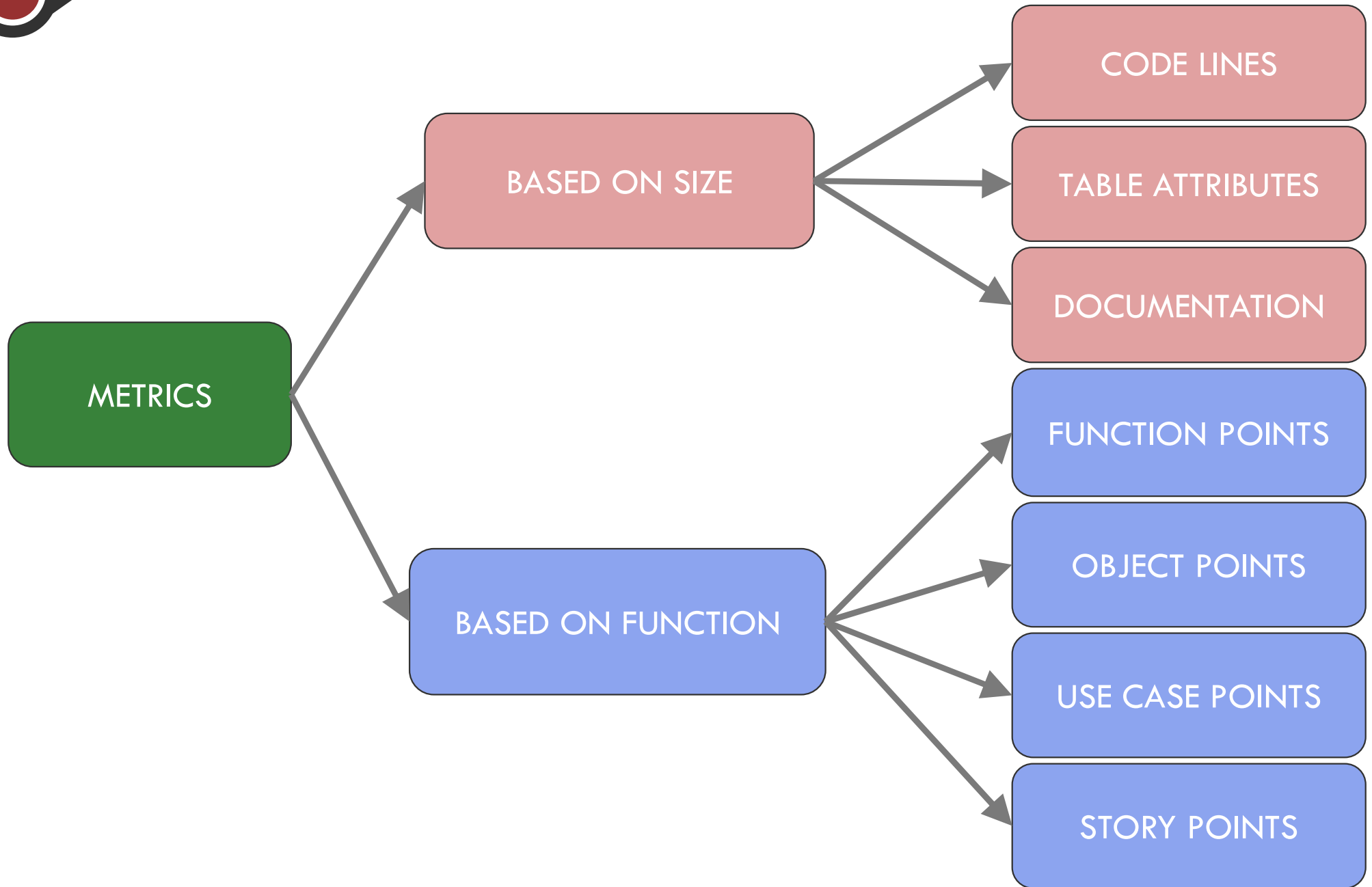
- Resources

Persons, salaries, productivity, experience, HW technical specifications...

Metrics

SW metrics

21



Metrics

Size-based metrics

22

Measure a product according to the amount of a physical aspect or a software artifact (source code lines, instructions in the object code, number of pages in the system documentation, number of attributes in a table...)

```
import java.io.*;
import java.net.*;
import java.util.*;
import protection;

public class client {
    public void sendAuthentication(String user, String password) {
        OutputStream outStream = new DataOutputStream(
            new DataOutputStream(out));
        long t1 = (new Date()).getTime();
        double q1 = Math.random();
        byte[] protected1 = protection.encrypt(q1, t1);
        long t2 = (new Date()).getTime();
        double q2 = Math.random();
        byte[] protected2 = protection.encrypt(q2, t2);
        out.writeUTF(user);
        out.writeInt(protected1.length);
        out.write(protected2);
        out.flush();
    }
}

public static void main(String[] args) {
    String host = args[0];
    int port = 7999;
    String user = "John";
    String password = "shh";
    Socket s = new Socket(host, port);

    Client client = new Client(s);
    client.sendAuthentication(user, password);
}
```

Metrics

Size-based metrics

23

- Are used as an estimation variable and as a metric extracted from previous projects.
- Easily and directly evaluated by counting, for instance, the number of lines of code.
- What is a line of code? (I could write extremely long lines)
- Depend on the language and the programming style.
- Difficult to predict for a given function.

Metrics

Function-based metrics

24

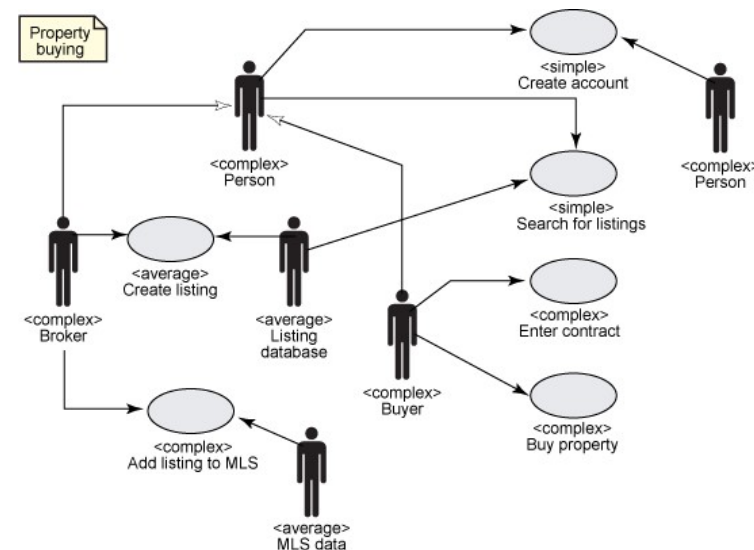
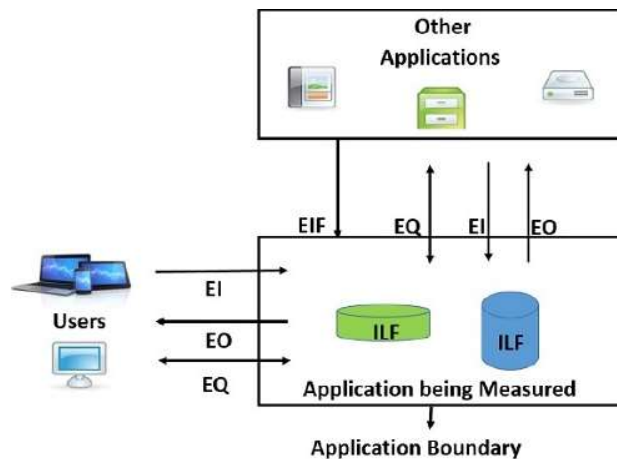
"Amount" of useful functionality generated:

Function points (inputs, outputs, queries, registers...)

Object points (screens, reports, complexity...)

Use case points (actors, UC, environmental and technical factors...)

Story points (complexity, speed...)



Metrics

Function-based metrics

25

- Can be applied from early development stages.
- Independent of the language, tools or methodologies.
- More subjective.

Estimation techniques

Classification

26

- Expert judgement
- Heuristic/Historical/Analogous estimating
- Parametric or algorithmic estimation
- Decomposition technique
- Phased estimation
- Pricing to win
- Delphi technique
- Other techniques

Estimation techniques

Expert judgement



27

1. Several experts provide individual estimates.
2. These estimates are compared and analyzed.
3. The process is repeated until a satisfactory result is achieved.

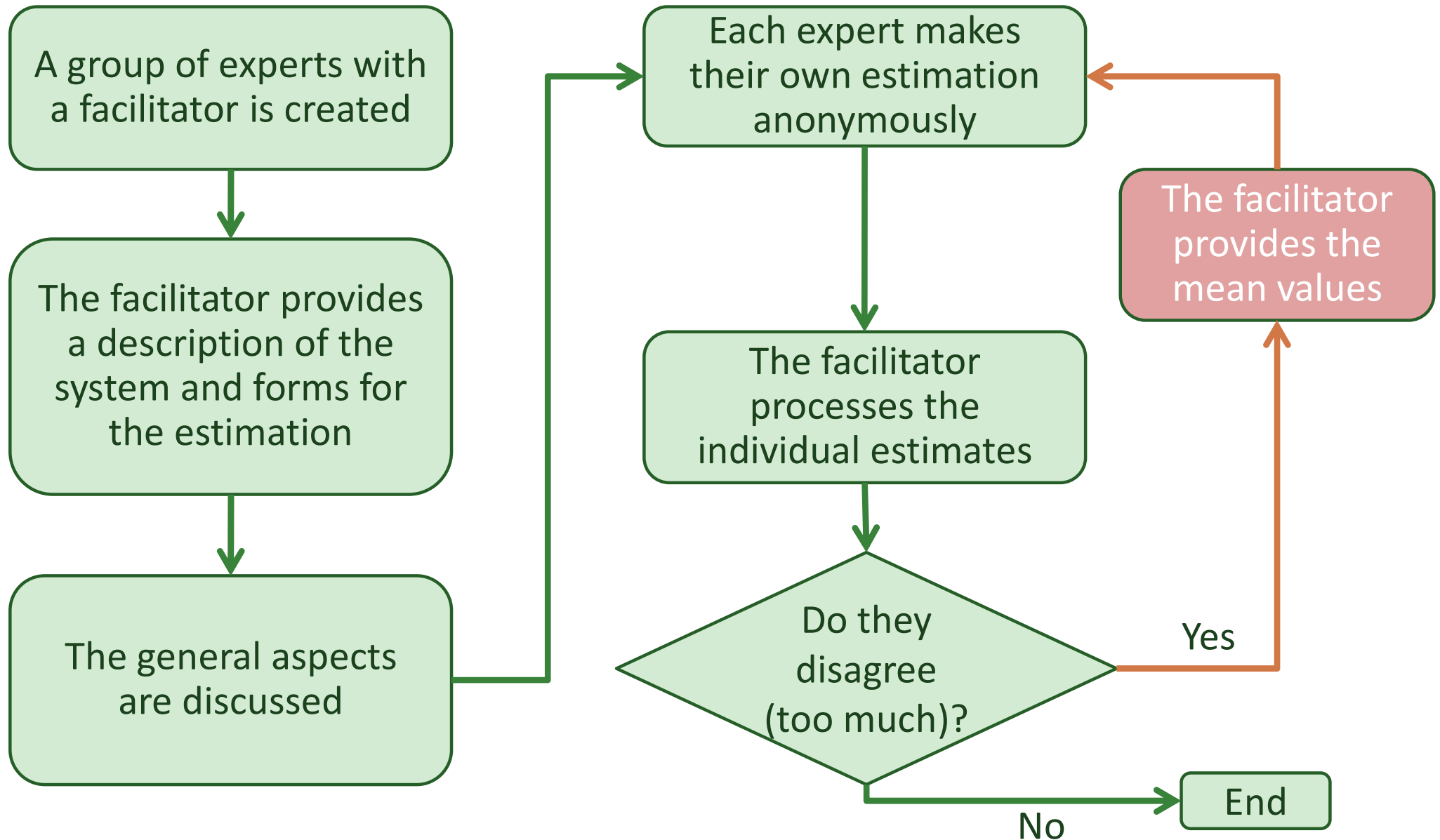
Pros and cons

- (+) Experts take into account some "subjective" factors which are not considered with other methods.
- (-) Experts are required.
- (-) Not reproducible.

Estimation techniques

Delphi technique/method

28



Estimation techniques

Heuristic/historical/analogous



29

Based on similarity, historical data or experience

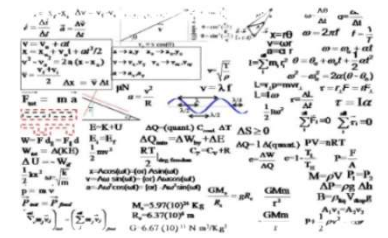
1. Reliable data from previous projects are available.
2. Project attributes are identified.
3. Projects with similar attributes are selected.
4. Estimates are adapted to the current project.
5. Experts are required for the previous steps.

Pros and cons

- (+) Based on past experience, which includes subjective factors.
- (-) Requires similar past projects.
- (-) How is similarity measured?
- (-) What was valid in the past might not be valid now.

Estimation techniques

Algorithmic or parametric



30

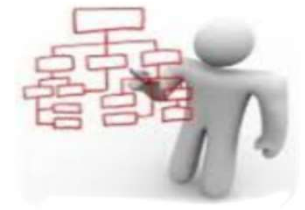
1. Based on predictive functions.
2. These functions are obtained by fitting with different parameters from previous projects.

Pros and cons

- (+) Reproducible calculations which can be refined.
- (+) The problem is reduced to the estimation of certain parameters.
- (+) Based on past experience with statistical significance.
- (-) What was valid in the past might not be valid now.
- (-) Difficult to adapt to exceptional circumstances or events.

Estimation techniques

Decomposition



31

1. An estimate is given for each lower level unit.
2. Estimates are more detailed.

Pros and cons

- (+) Quite exact.
- (-) Arduous.

Estimation techniques

Phased



32

1. Estimate the project phase by phase.
2. The estimation is more complete and detailed for the next phase.
3. A more general estimate is given for the rest of phases.
4. Incorporates the re-estimation.

Pros and cons

- (+) Useful when risks are high.
- (-) Does not provide estimates for the whole project from the beginning.



1. The client has a limited budget.
2. The client cannot provide a clear and detailed description of what they want.
3. The price of the product is set to the amount the client can pay.
4. The software requirements are identified.
5. The requirements are implemented in such a way that they match the price.
6. Some requirements can be left undone.

Pros and cons

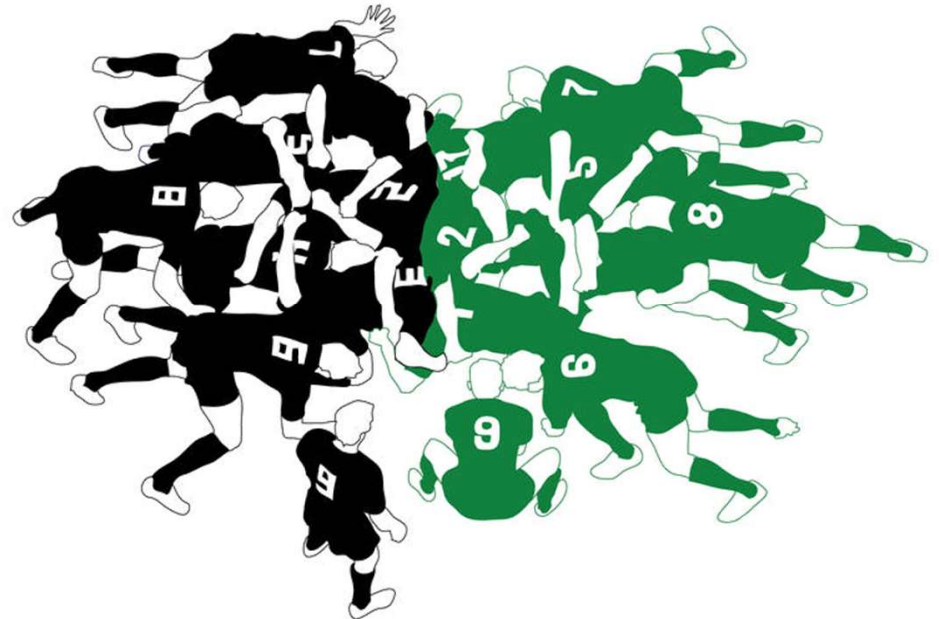
- (+) The price does not exceed the client's limit.
- (-) Some requirements might not be implemented.

Scrum

Principles

34

- Iterative
- Incremental development
- Agile approach
- Open to volatile requirements
- Adaptable to changes



Estimation of user stories

35

- Estimate stories in story points, which are relative estimates of the complexity, effort or duration of a story.
- Estimating stories needs to be done by the team, and the estimates are owned by the team rather than individuals.
- Triangulate an estimate by comparing it to other estimates.
- Whether or not a team programs in pairs has no impact on story point estimates. Pair programming affects the team's velocity, not their estimates.

Estimation of user stories

36

Developer Responsibilities

- Defining story points in a manner that is relevant and usable by your team and consistently sticking to that definition.
- Giving honest estimates and not giving in to temptation or pressure to give low estimates.
- Estimating as a team.
- Giving estimates that are consistent with other estimates (e.g., all the two-point stories should be similar).

Estimation of user stories

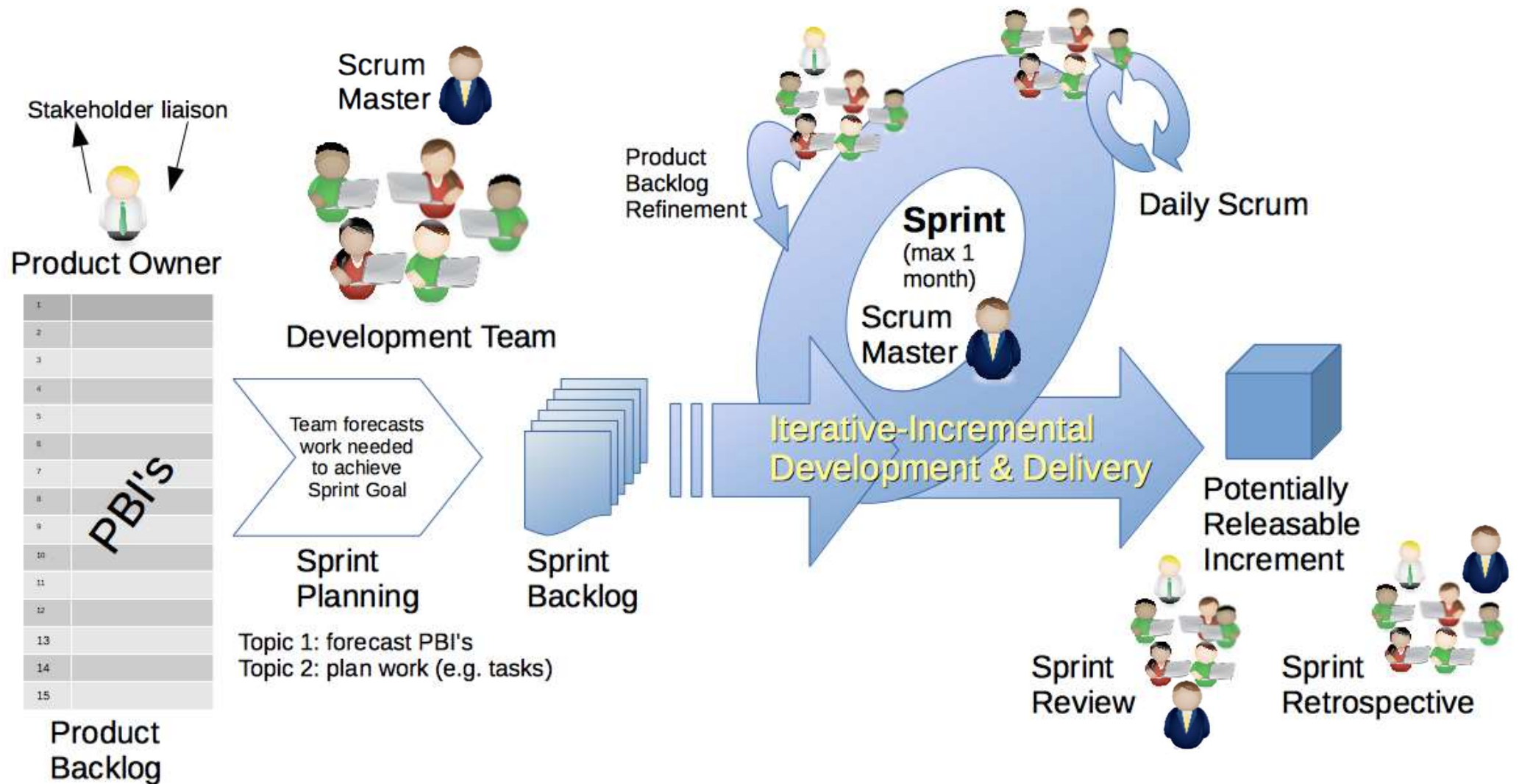
37

Customer Responsibilities

- Participating in estimation meetings, answering questions and clarifying stories, not estimating stories yourself.

Scrum Scheme

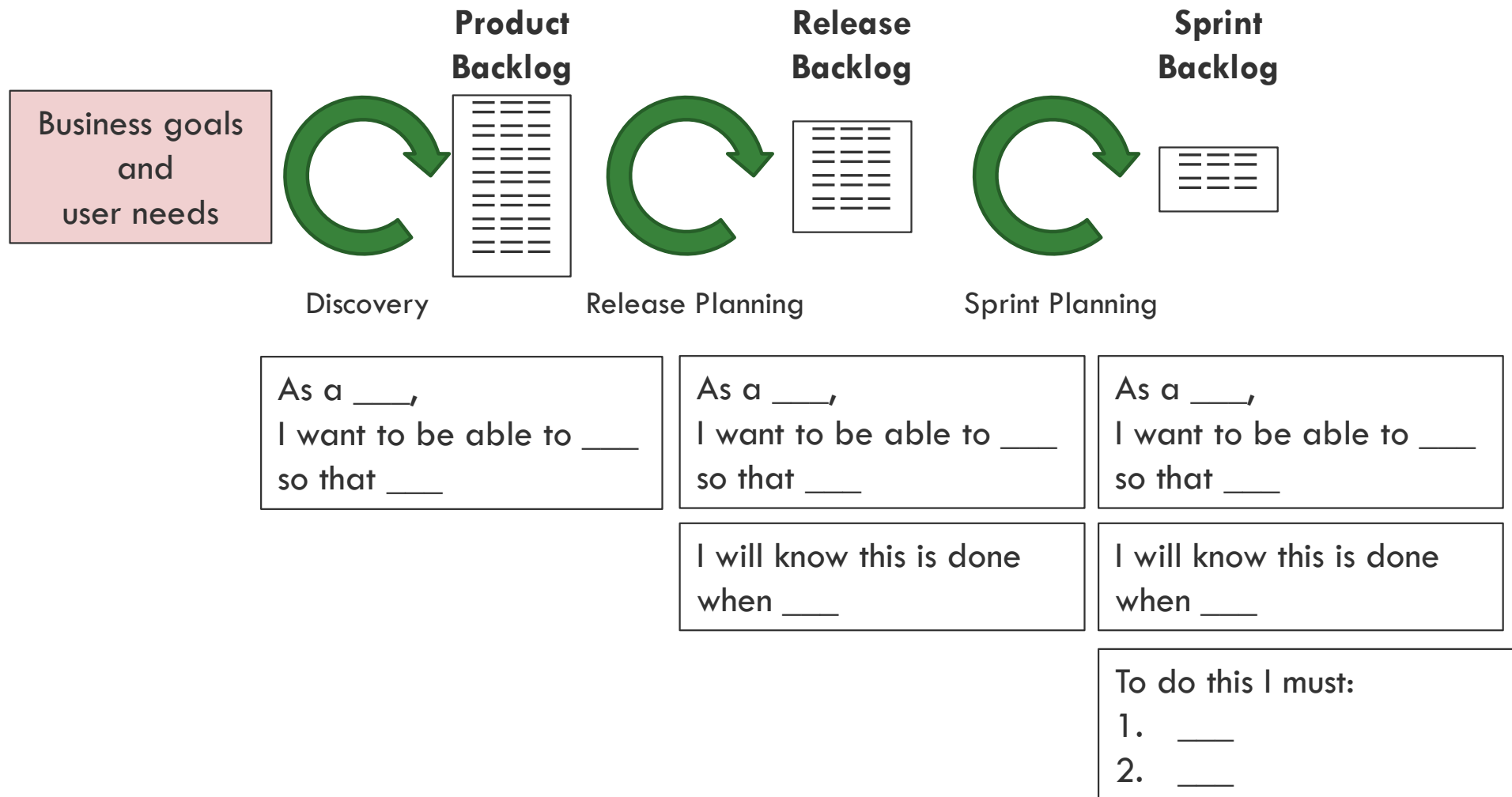
38



User stories

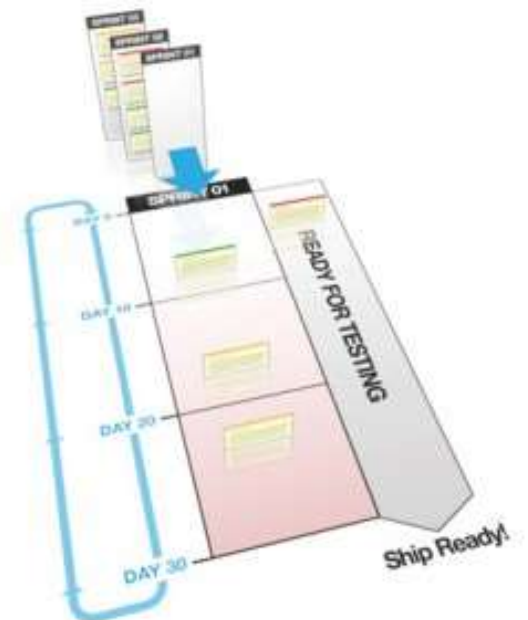
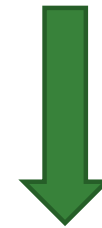
Life cycle

39



Life-cycle with several releases

40



Themes, epics, stories and tasks

41



Estimation techniques

Planning Poker (Scrum Poker)

42



Estimation techniques

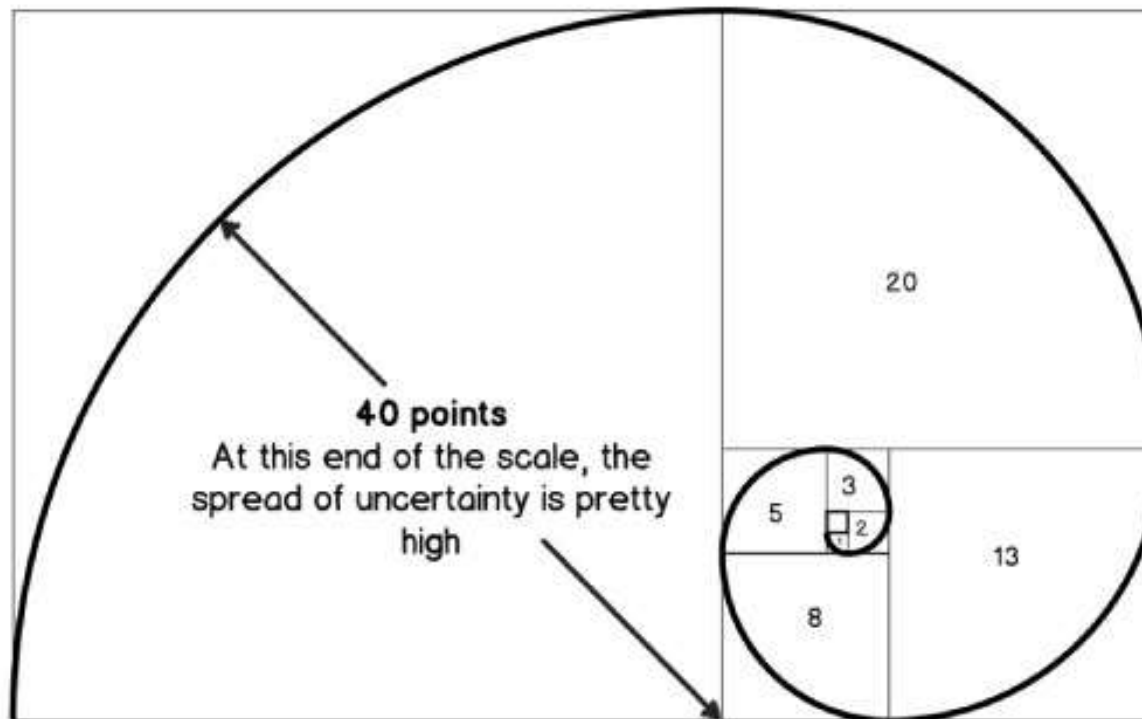
Planning Poker (Scrum Poker)

43

Fibonacci:



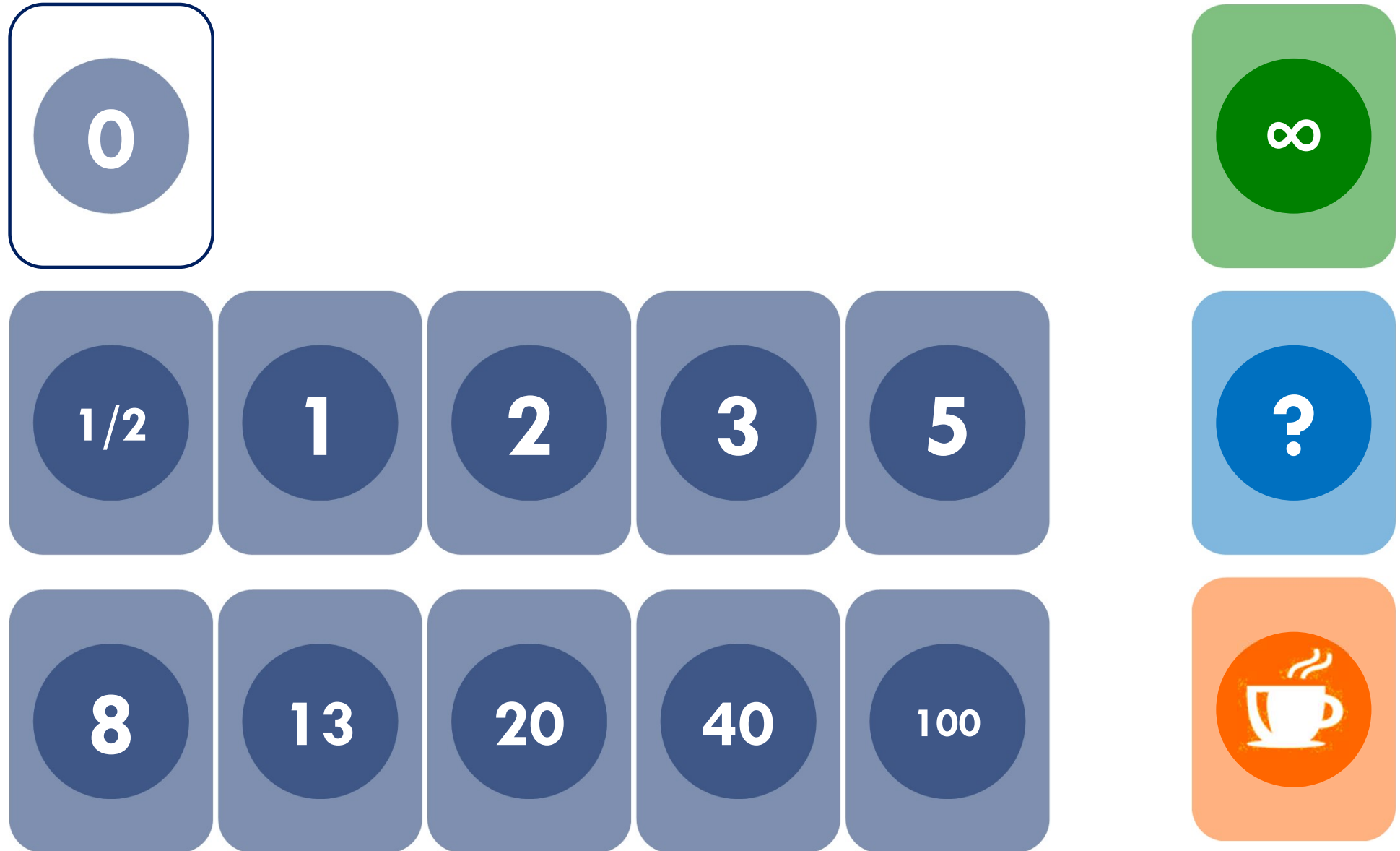
Scrum Poker:



Estimation techniques

Planning Poker (Scrum Poker)

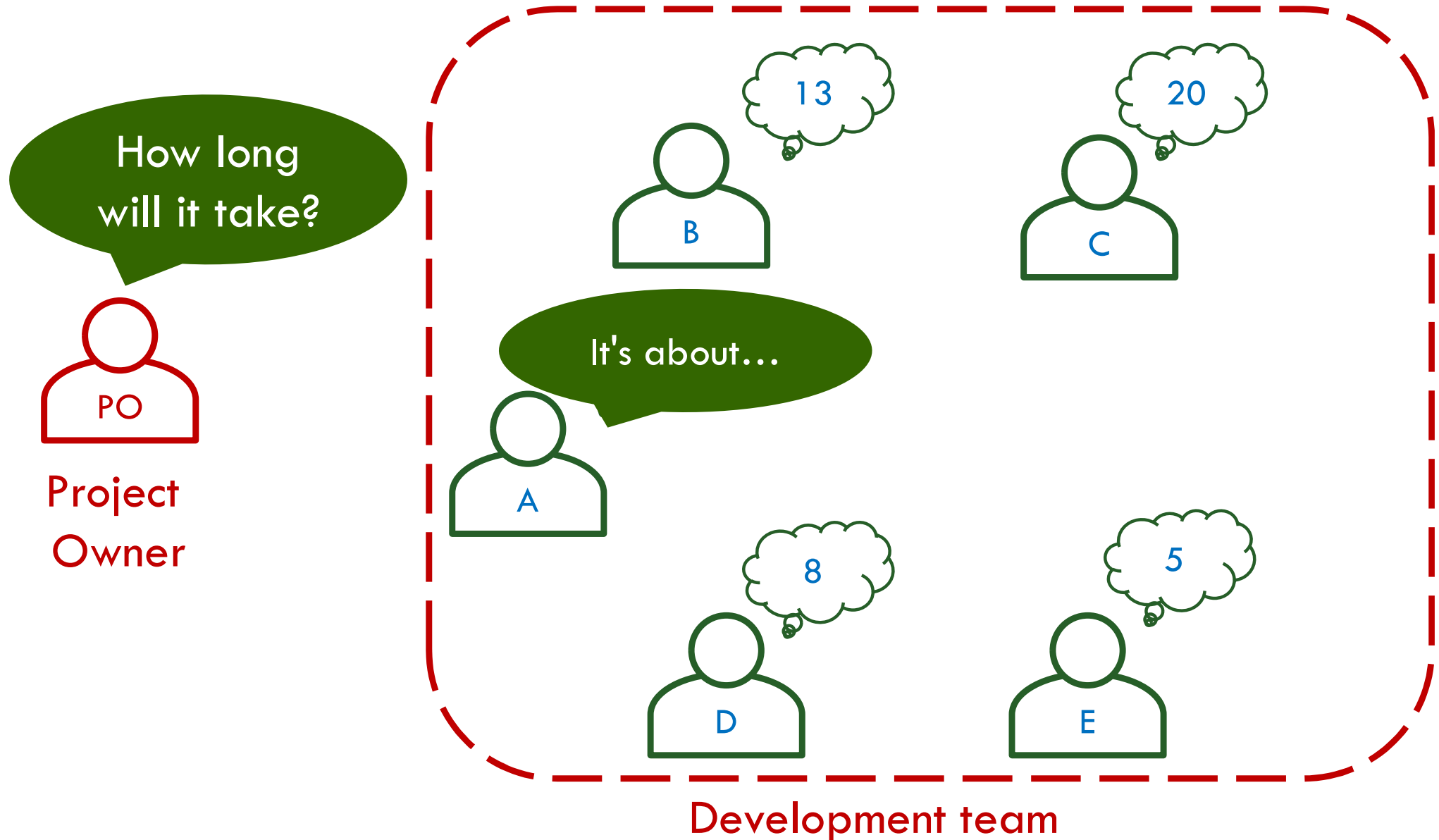
44



Estimation techniques

Planning Poker (Scrum Poker)

45



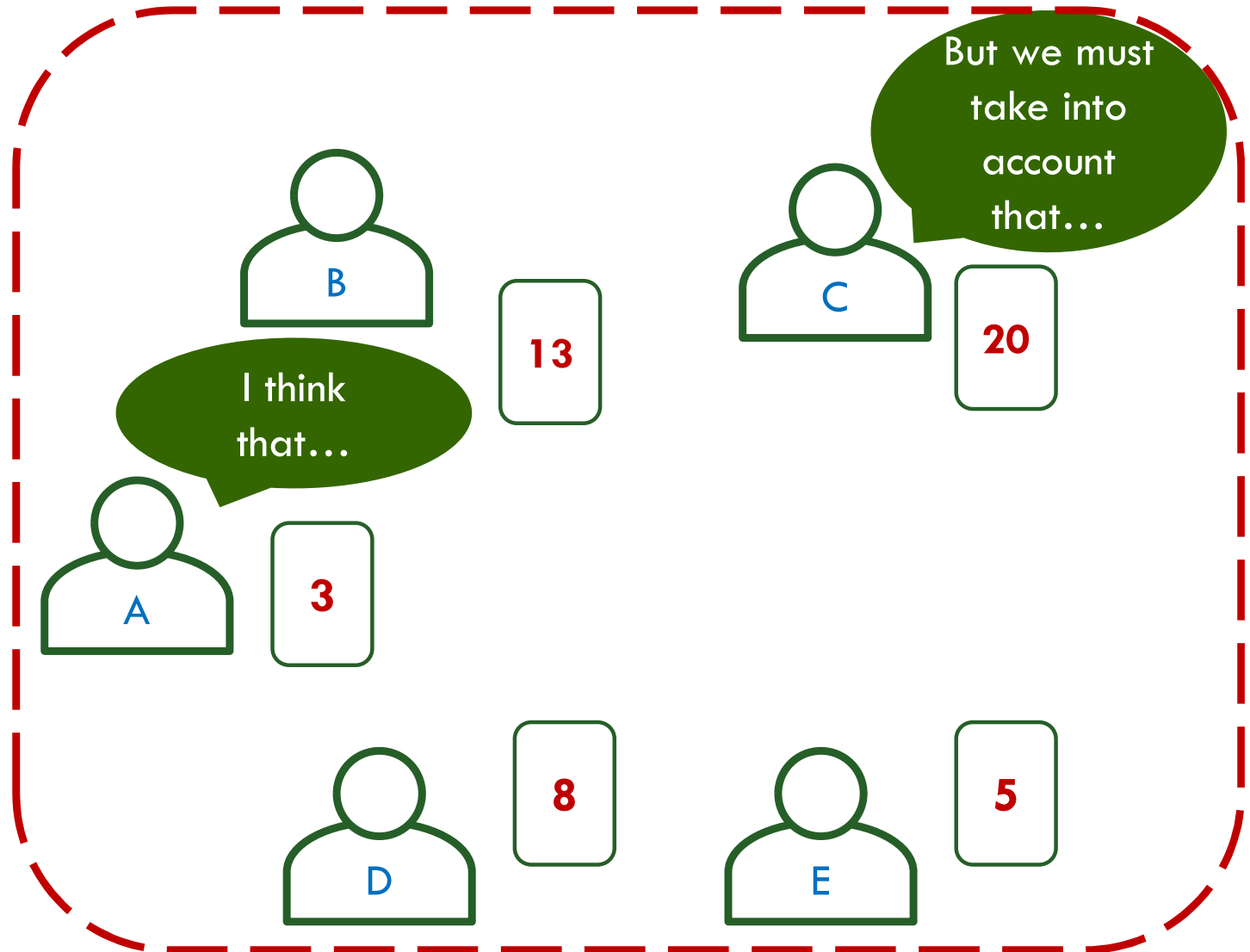
Estimation techniques

Planning Poker (Scrum Poker)

46



Project
Owner



Development team

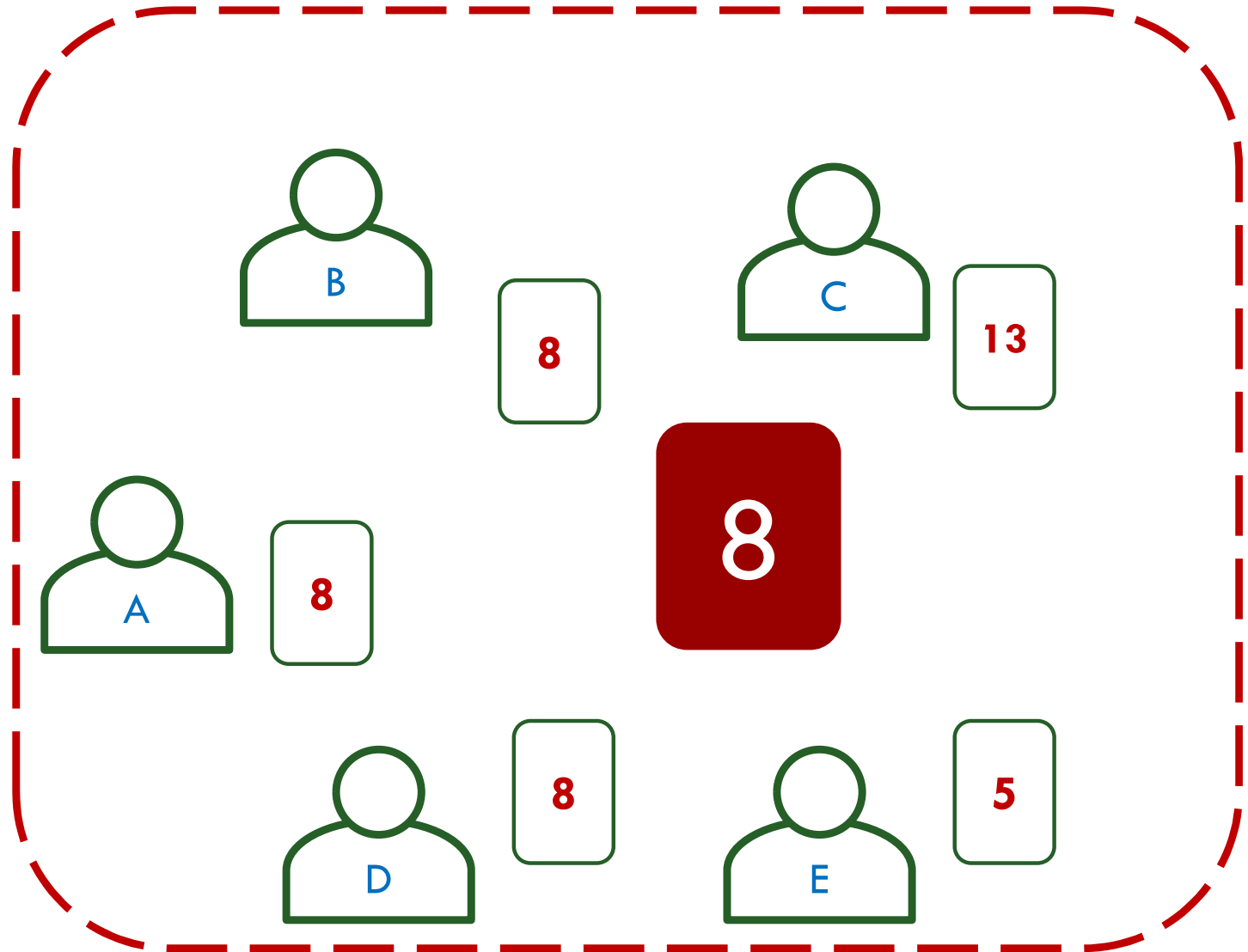
Estimation techniques

Planning Poker (Scrum Poker)

47



Project
Owner

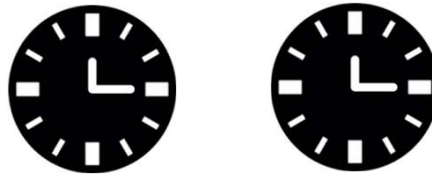


Development team

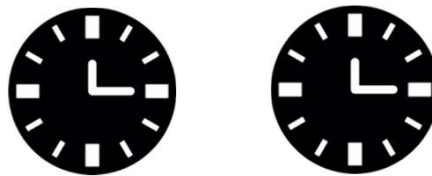
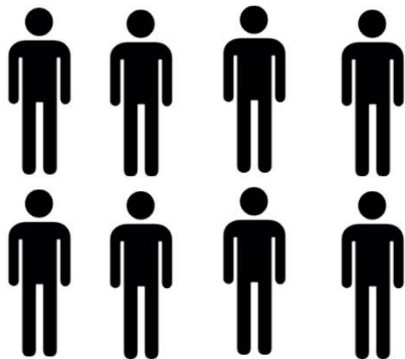
Estimating user stories

48

Team A finished 43 story points in their last two-week iteration.



Team B is working on a separate project and has twice as many developers. They also completed 43 story points in their last two-week iteration.



How can that be?

Estimating user stories

49

- Actual days will differ from ideal days
- It is common to start with an expected velocity that is somewhere between one-third and one-half the number of developer-days in an iteration.

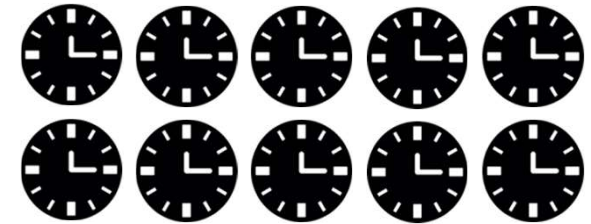
Estimating user stories

50

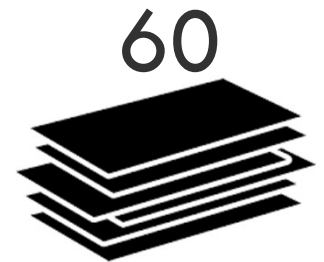
For example,
a six-person team



using two-week
(ten working day) iterations



would have sixty developer-days in each iteration.



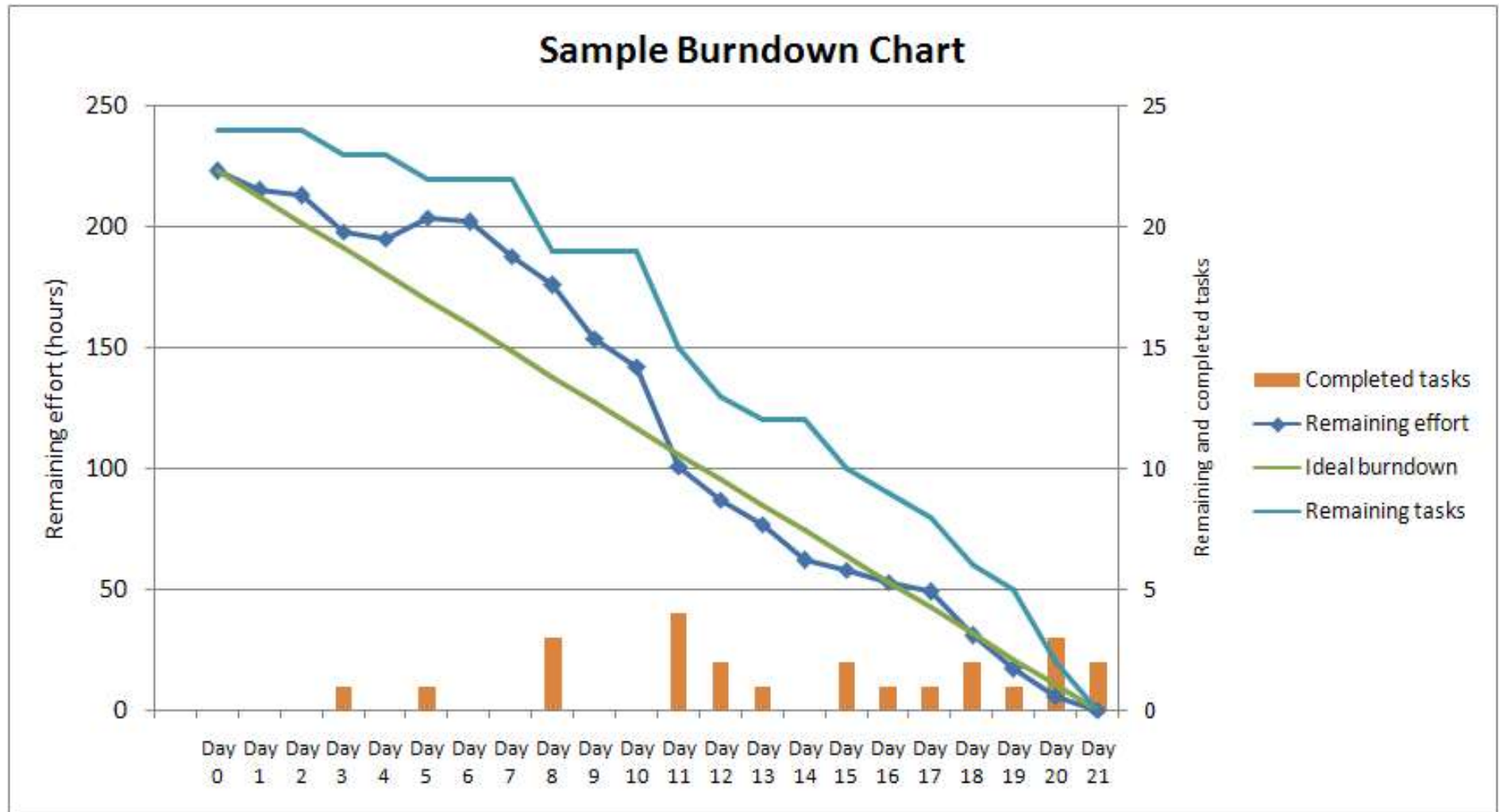
They may want to estimate velocity
between 20 and 30 story points per iteration

20-30



Burndown Charts

51



Example

Online Bookshop

52



Example

Online Bookshop

53

<i>A user can do a basic simple search that searches for a word or phrase in both the author and title fields.</i>	1	Must have
<i>A user can search for books by entering values in any combination of author, title and ISBN.</i>	1	Should have
<i>A user can view detailed information on a book. For example, number of pages, publication date and a brief description.</i>	1	Could have
<i>A user can put books into a "shopping cart" and buy them when they are done shopping.</i>	1	Won't have
<i>A user can remove books from their cart before completing an order.</i>	½	
<i>To buy a book the user enters their billing address, the shipping address and credit card information.</i>	2	
<i>A user can rate books from 1 (bad) to 5 (good). the book does not have to be one the user bought from us.</i>	2	
<i>A user can write a review of a book. They can preview the review before submitting it. The book does not have to be one the user bought from us.</i>	5	
<i>An administrator needs to approve or reject reviews before they are available on the site.</i>	2	

Example

Online Bookshop

54

<i>A user can establish an account that remembers shipping and billing information.</i>	2	Must have
<i>A user can edit the credit card information stored in their account.</i>	½	Should have
<i>A user can edit the shipping and billing addresses stored in their account.</i>	1	Should have
<i>A user can put books into a "wish list" that is visible to other site visitors.</i>	2	Could have
<i>A user, especially a Non-Sailing Gift Buyer, can search for a wish list based on its owner's name and state.</i>	1	Won't have
<i>A user can check the status of their recent orders.</i>	½	Could have
<i>If an order has not shipped, a user can add or remove books, change the shipping method, the delivery address and the credit card.</i>	1	Won't have
<i>A user can place an item from a wish list (even someone else's) into their shopping cart.</i>	½	Could have
<i>A repeat customer must be able to find one book and complete an order in less than 90 seconds.</i>	0	Won't have

Example

Online Bookshop

55

A user can view a history of all of their past orders.	1	Must have
A user can easily re-purchase items when viewing past orders.	½	Should have
The site always tells a shopper what the last 3 (?) items they viewed are and provides links back to them (this works even between sessions).	1	Could have
A user can see what books we recommend on a variety of topics.	3	Won't have
A user can choose to have items gift wrapped.	½	
A user can choose to enclose a gift card and can write their own message for the card.	½	
A Report Viewer can see reports of daily purchases broken down by book category, traffic, best- and worst-selling books and so on.	8	
A user must be properly authenticated before viewing reports.	1	
Orders made on the website have to end up in the same order database as telephone orders.	0	

Example

Online Bookshop

56

An administrator can add new books to the site.

1

An administrator can delete a book.

½

An administrator can edit the informaion about an existing books.

1

The system must support peak usage of up to 50 concurrent users.

0

Must have

Should have

Could have

Won't have

References

57

“User Stories Applied
For agile software development”
Mike Crohn
The Addison Wesley signature series
2009