System Development and Project Organization (BSUP)

Paolo Tell

Scrum

a detailed overview

Mountain Goat Software, LLC

IT University of Copenhagen

Outline

- Literature
  - "The New New Product Development Game"
- Scrum in details
  - Roles
  - Cerimonies
  - Artifacts
- User stories in details

Roles

- Product owner
- ScrumMaster
- Team

Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

Ceremonies

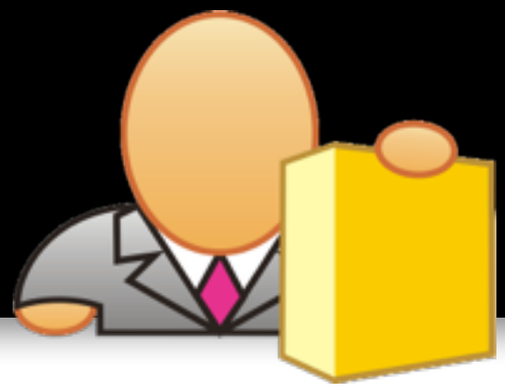
- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Scrum framework - Roles

The Product Owner



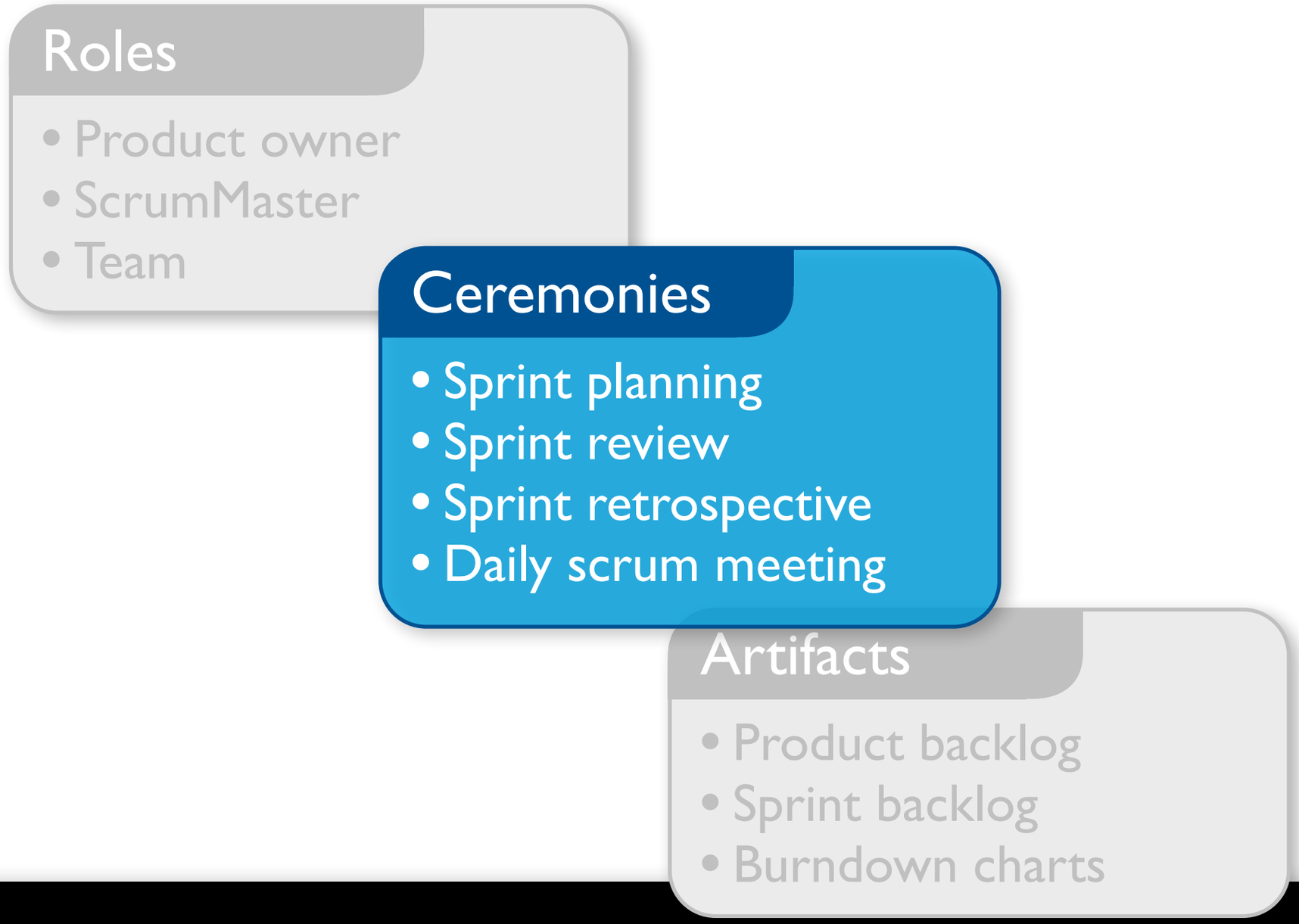
- Define the features of the product
- Decide on release date and content
- Be responsible for the profitability of the product (ROI)
- Prioritize features according to market value
- Adjust features and priority every iteration, as needed
- Accept or reject work results

The ScrumMaster

- Represents management to the project
- Responsible for enacting Scrum values and practices
- Removes impediments
- Ensure that the team is fully functional and productive
- Enable close cooperation across all roles and functions
- Shield the team from external interferences

The Team

- Typically 5-9 people
- Cross-functional:
  - Programmers, testers, user experience designers, etc.
- Members should be full-time
  - May be exceptions (e.g., database administrator)
- Teams are self-organizing
  - Ideally, no titles but rarely a possibility
- Membership should change only between sprints



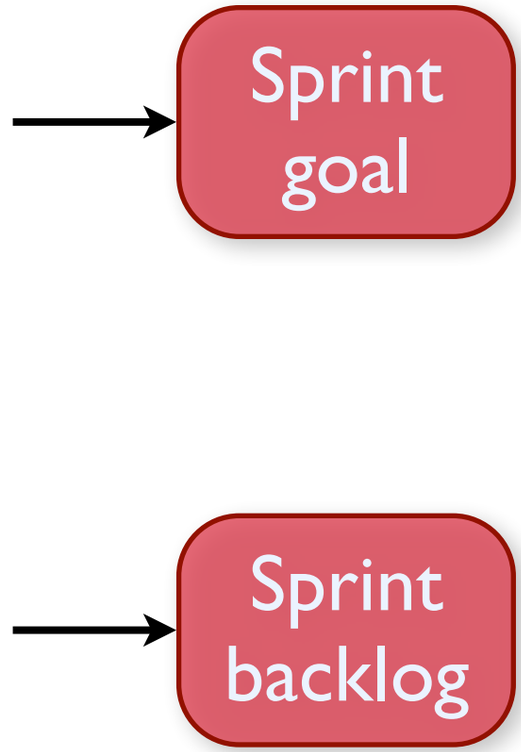
Scrum framework - Ceremonies

The sprint planning meeting

- Who
  - Team, ScrumMaster, and Product owner
- Agenda
  - Discuss top priority product backlog items
  - The team selects which to do
- Why
  - Know what will be worked on
  - Understand it enough to do it

The sprint planning meeting

- Who
  - Team, ScrumMaster, and Product owner
- Agenda
  - Discuss top priority product backlog items
  - The team selects which to do
- Why
  - Know what will be worked on
  - Understand it enough to do it



The sprint planning meeting

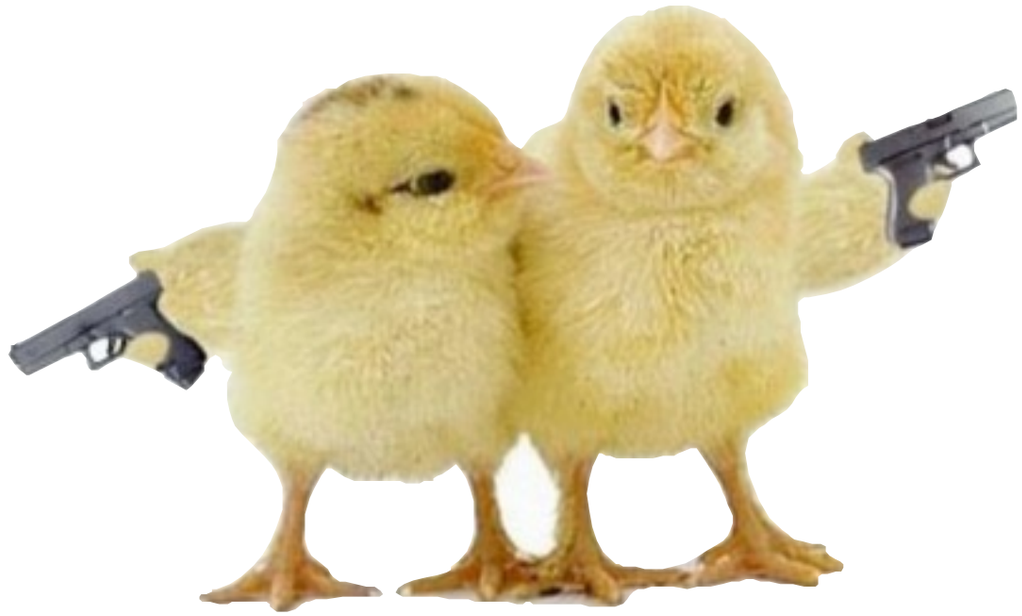
As a vacation planner,
I want to see photos
of the hotels.

Code the middle tier (8 hours)
Code the user interface (4)
Write test fixtures (4)
Code the foo class (6)
Update performance tests (4)

- Team selects items from the product backlog they can commit to completing
- Sprint backlog is created
  - Tasks are identified and each is estimated (1-16 hours)
  - Collaboratively, not done alone by the ScrumMaster
- High-level design is considered

The daily scrum meeting

- Parameters
  - Daily
  - 15-minutes
  - Stand-up



- Not for problem solving
  - Whole world is invited
  - Only team members, ScrumMaster, product owner, can talk

- Helps avoid other unnecessary meetings

Everyone answers 3 questions

1. What did you do yesterday?
2. What will you do today?
3. Is anything in your way?

- These are <u>not</u> status for the ScrumMaster
  - They are commitments in front of peers
  - They are synchronization meetings

The sprint review meeting

- Team presents what it accomplished during the sprint
- Typically takes the form of a demo of new features or underlying architecture
- Informal
  - 2-hour prep time rule
  - No slides
- Whole team participates
- Invite the world

The sprint retrospective meeting

- Periodically take a look at what is and is not working
- Typically 15–30 minutes

- Done after every sprint

- Whole team participates
  - ScrumMaster
  - Product owner
  - Team

Start / Stop / Continue

- Whole team gathers and discusses what they'd like to:

Start doing

Stop doing

Continue doing

This is just one of many ways to do a sprint retrospective.

Roles

- Product owner
- ScrumMaster
- Team

Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Scrum framework - Artifacts

The product backlog

- The requirements
- A list of all desired work on the project
- Ideally expressed such that each item has value to the users or customers of the product
- Prioritized by the product owner
- Reprioritized at the start of each sprint

This is the product backlog

A sample product backlog

Backlog item	Estimate
Allow a guest to make a reservation	3
As a guest, I want to cancel a reservation.	5
As a guest, I want to change the dates of a reservation.	3
As a hotel employee, I can run RevPAR reports (revenue-per-available-room)	8
Improve exception handling	8
...	30
...	50

The sprint goal

Life Sciences

Support features necessary for population genetics studies.

Database Application

Make the application run on SQL Server in addition to Oracle.

Financial services

Support more technical indicators than company ABC with real-time, streaming data.

- A short statement of what the work will be focused on during the sprint

Managing the sprint backlog

- Individuals sign up for work of their own choosing
  - Work is never assigned

- Estimated work remaining is updated daily

- Any team member can add, delete or change the sprint backlog

- Work for the sprint emerges

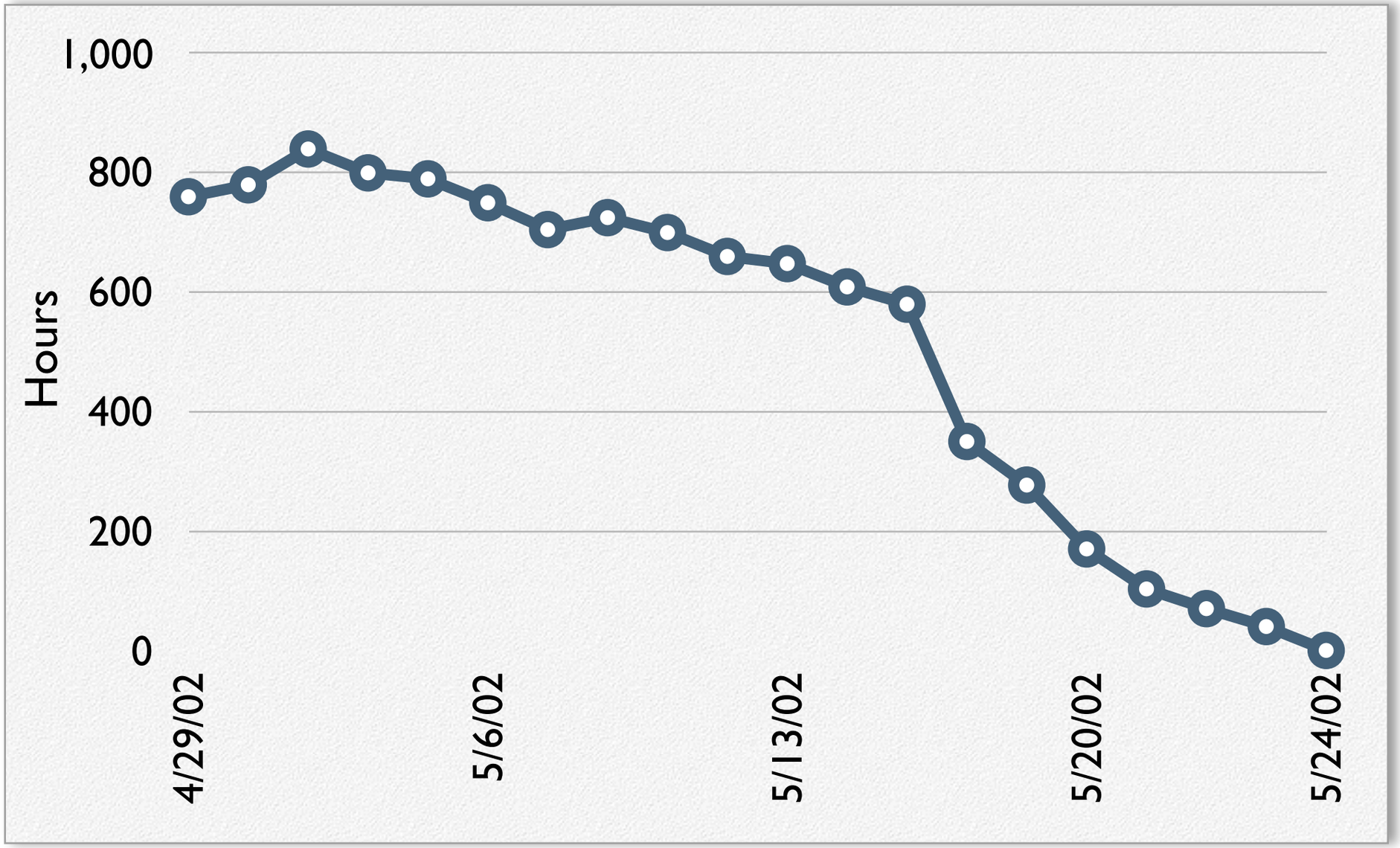
- If work is unclear, define a sprint backlog item with a larger amount of time and break it down later

- Update work remaining as more becomes known

A sample sprint backlog

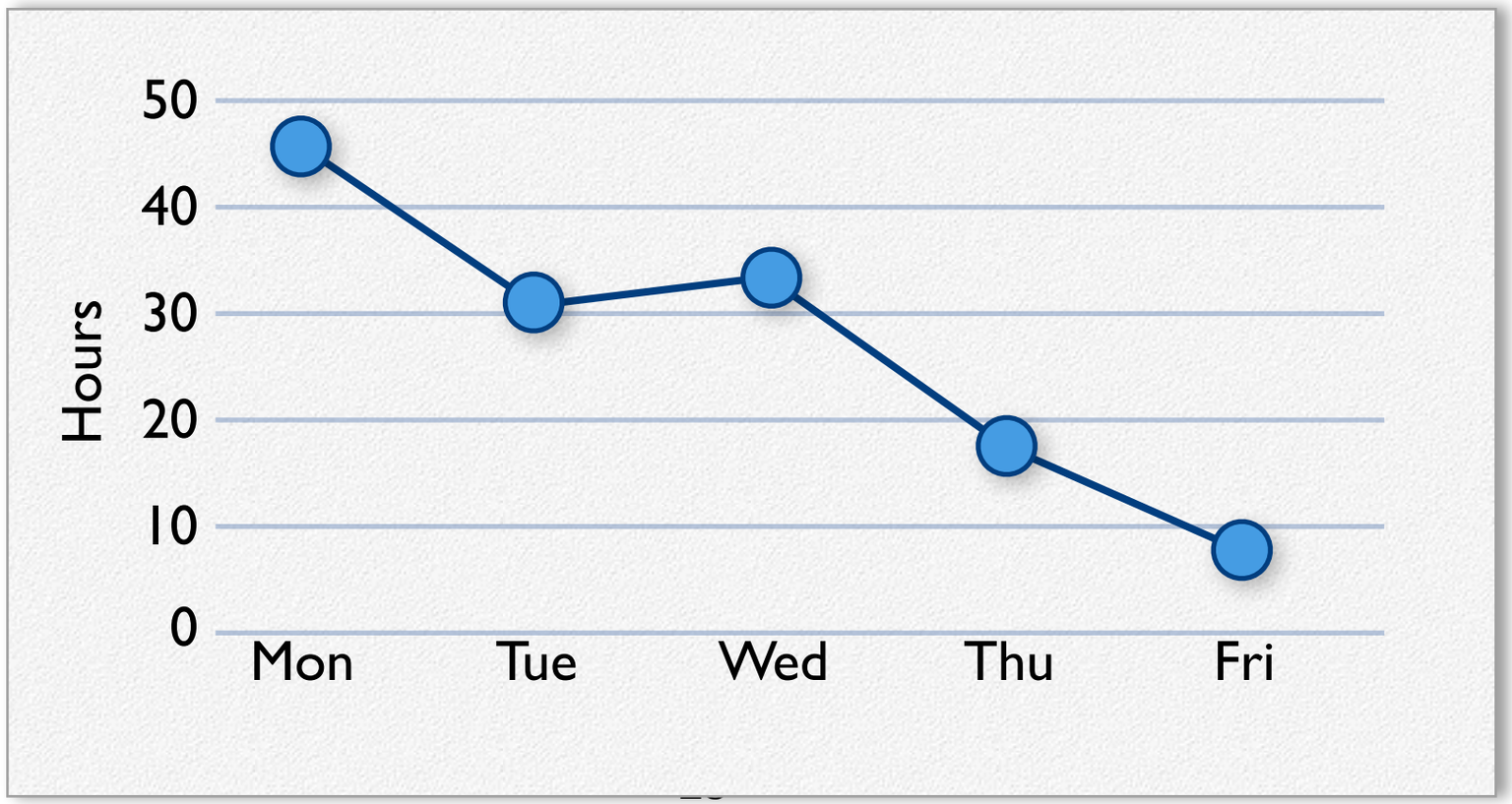
Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	8
Write online help	12				
Write the foo class	8	8	8	8	8
Add error logging			8	4	

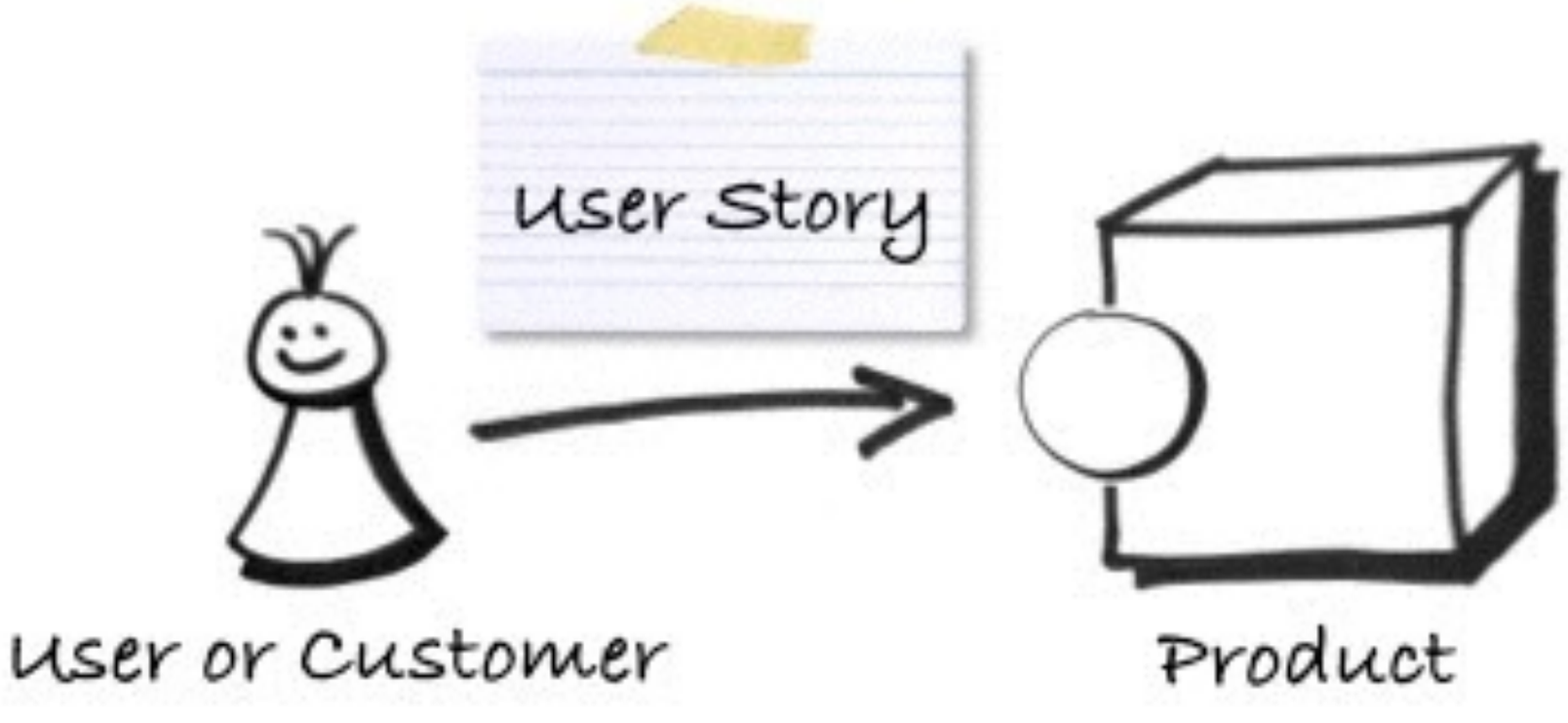
A sprint burndown chart



A sprint burndown chart

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	7	
Test the middle tier	8	16	16	11	8
Write online help	12				





User Stories

What problem do stories address?

- Software requirements is a communication problem

- Those who want the software must communicate with those who will build it

Balance is critical

- If either side dominates, the business loses

- If the business side dominates …
  - … functionality and dates are mandated with little regard for reality whether the developers understand the requirements
  - "I need this, I need it by this date, make it happen"

- If the developers dominate …
  - … technical jargon replaces the language of the business and developers lose the opportunity to learn from listening
  - "we will build whatever you want, but you have to fill out this complicated use-case template. Fill this out and that will tell us what to build for you. If you do not fill this out, we will not build you the right thing"

Resource allocation

- We need a way of working together so that resource allocation becomes a shared problem

- Project fails when the problem of resource allocation falls too far from one side

Responsibility for resource allocation

- If the developers responsible …
  - … may trade quality for additional features
  - … may only partially implement a feature
  - … may solely make decisions that should involve the business

- If the business is responsible …
  - … lengthy upfront requirements negotiation and signify
  - … features are progressively dropped and the deadline nears

Imperfect schedules

- We cannot perfectly predict a software schedule
  - As users see the software, they come up with new ideas
  - Too many intangibles
  - Developers have a notoriously hard time estimating

- If we can't perfectly predict a schedule, we can't perfectly say what will be delivered

So what do we do?

This is where user stories come in

- We make decisions based on the information we have
  - … but do it often

- Rather then making one all-encompassing set of decisions
  - … we spread decision-making across the project

Agenda

- [ ] What stories are
- [ ] Writing user stories
- [ ] Why user stories

Agenda

- [x] What stories are
- [ ] Writing user stories
- [ ] Why user stories

Three Cs

Card

- Stories are traditionally written on note cards.
- Cards may be annotated with estimates, notes, etc.

Conversation

- Details behind the story come out during conversations with the product owner

Confirmation

- Acceptance tests confirm a story was coded correctly

Samples from a travel website

As a user, I want to reserve a hotel room.

As a vacation traveler, I want to see photos of the hotels.

As a user, I want cancel a reservation.

As a frequent traveler, I want to rebook a past trip so that I save time booking trips I take often.

Where are the details?

As a user, I want cancel a reservation.

- Does the user get a full or partial refund?
  - Is that refund to get credit card or is it site credit?
- How far ahead must the reservation be cancelled?
  - Is that the same for all hotels?
  - For all site visitors? Can frequent travelers cancel later?
- Is a confirmation provided to the user?
- How?

Details as conditions of satisfaction

As a user, I want cancel a reservation.

- ☐ Verify that a premium member can cancel the same day without a fee.
- ☐ Verify that a non-premium member is charged 5% for the same-day cancellation.
- ☐ Verify that a confirmation email is sent.
- ☐ Verify that the hotel is notify of any cancellation.

- The product owner's conditions of satisfaction can be added to a story
- There are essentially tests

Details added in smaller sub-stories

As a user, I want cancel a reservation.

As a premium member, I can cancel a reservation up to the last minute.

As a non-premium member, I can cancel but I am charged 5%.

As a hotel, I am emailed a notification of any cancelled reservation.

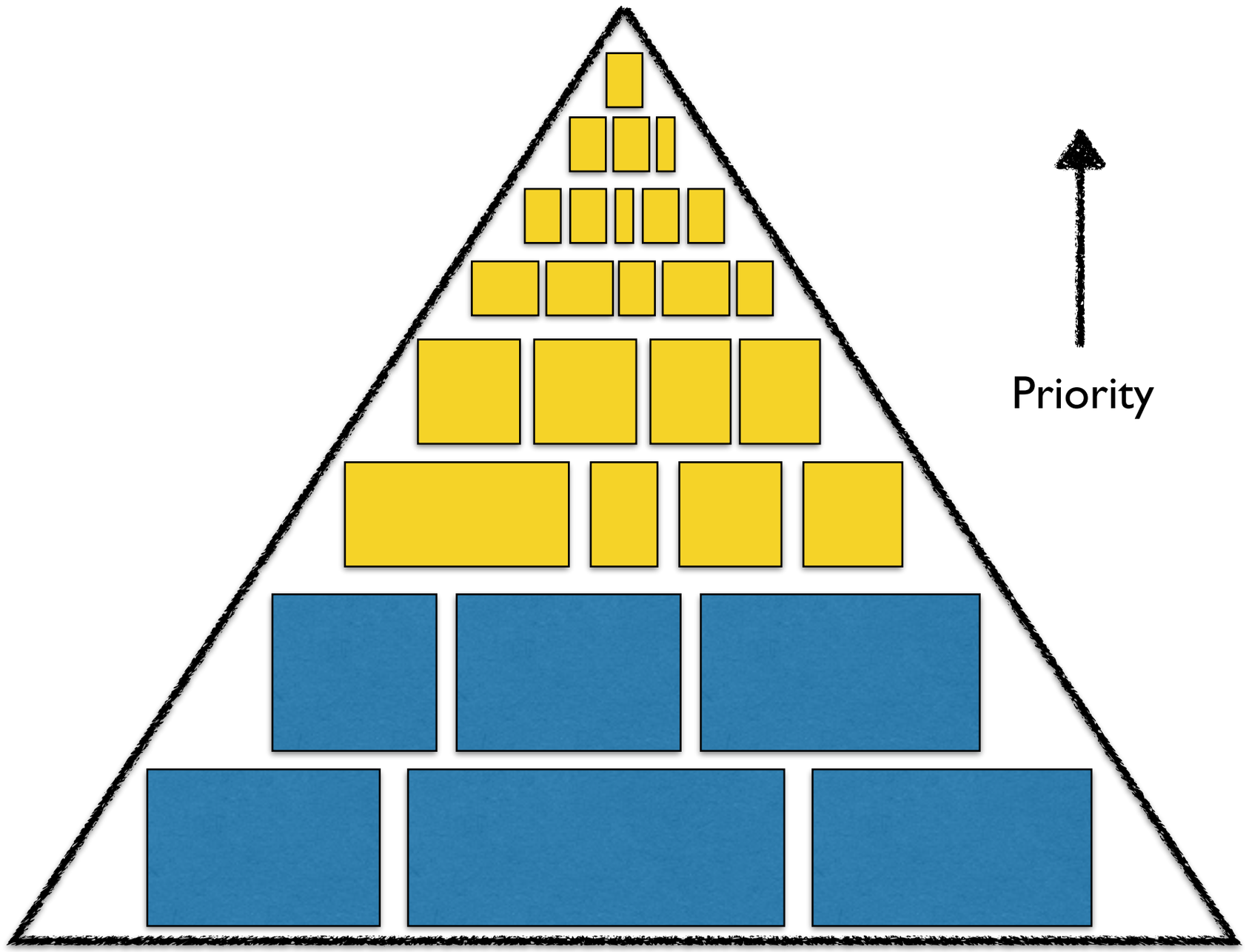
Techniques can be combined

- These approaches are not mutually exclusive

- Write stories at an appropriate level

- By the time it's implemented, each story will have conditions of satisfactions associated with it

The product backlog iceberg



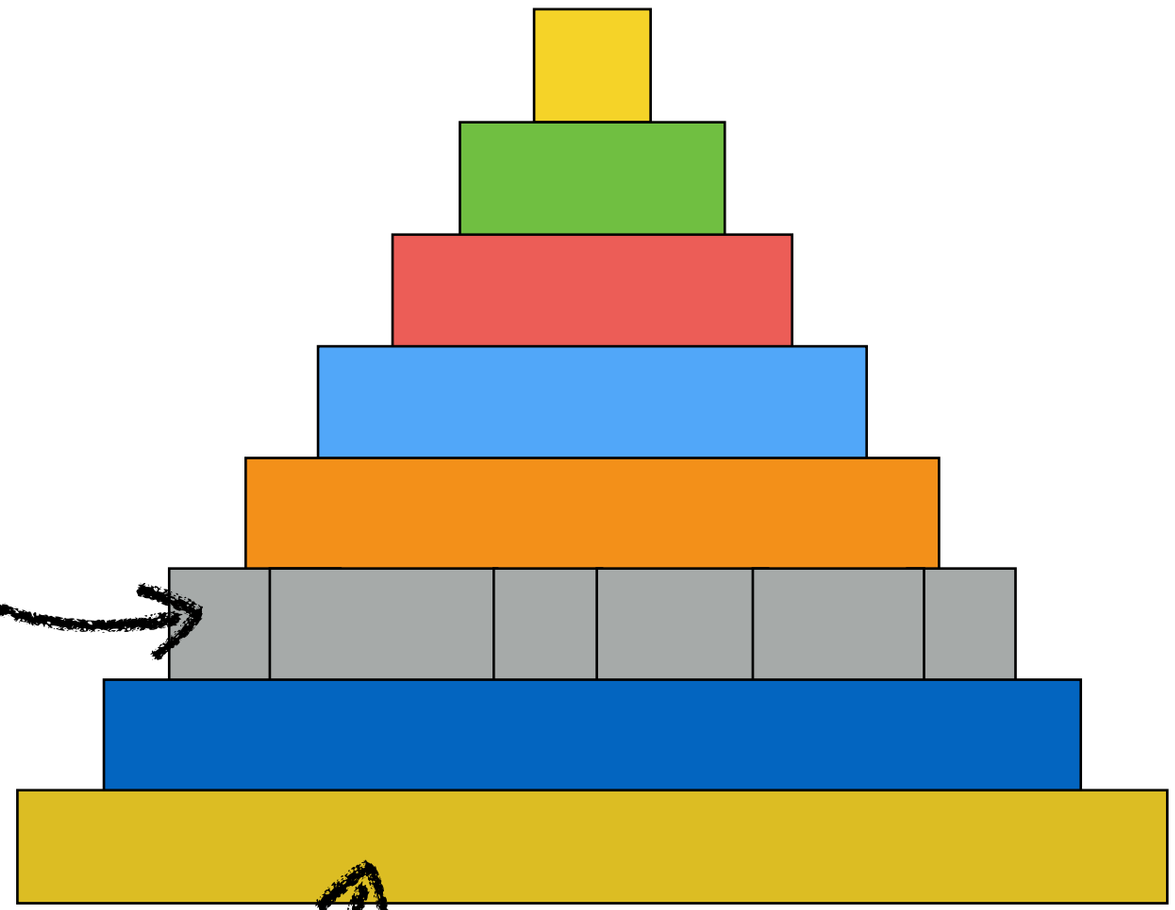
Some additional useful terms

Theme

- A collection of related user stories

Epic/Saga

- A large user story



An example

As a VP Marketing, I want to review the performance of historical promotional campaigns so that I can identify and repeat profitable ones.

Clearly an epic

As a VP Marketing, I want to select the timeframe to use when reviewing the performance of past promotional campaigns, so that…

Epics???

As a VP Marketing, I can select which type of campaigns (direct mail, TV, email, radio, etc.) to include when reviewing the performance of past …

An example

As a VP Marketing, I want to see information on direct mailings when reviewing historical campaigns.

As a VP Marketing, I want to see information on TV ads when reviewing historical campaigns.

As a VP Marketing, I can select which type of campaigns (direct mail, TV, email, radio, etc.) to include when reviewing the performance of past …

As a VP Marketing, I want to see information on email ads when reviewing historical campaigns.

Agenda

- ☐ What stories are
- ☑ Writing user stories
- ☐ Why user stories

Exercise: logging in [5’]

“As a <user role>, I <want/need/can/etc> <goal> so that <reason>.”

- See how many user stories you can write about logging in.

- Examples:
  - As a registered user, I am required to log in so that I can access the system.
  - As a forgetful user, I can request a password remainder so that I can log in if I forget my mine.

Story writing workshops

- Includes whole team plus possibly some external stakeholders
- Typically not done every sprint
- Brainstorm to generate stories
- Goal is to write as many stories as possible
  - Some will be "implementation ready"
  - Others will be epics
- No prioritisation at this point

Start with epics and iterate

As a frequent flyer, I want to see check my account.

Frequent Flyer

As a frequent flyer, I want to book a trip.

As a frequent flyer, I want to …

Start with epics and iterate

Frequent Flyer

As a frequent flyer, I want to see check my account.

As a frequent flyer, I want to book a trip.

As a frequent flyer, I want to …

As a frequent flyer, I want to book a trip using miles.

As a frequent flyer, I want to rebook a trip I take often.

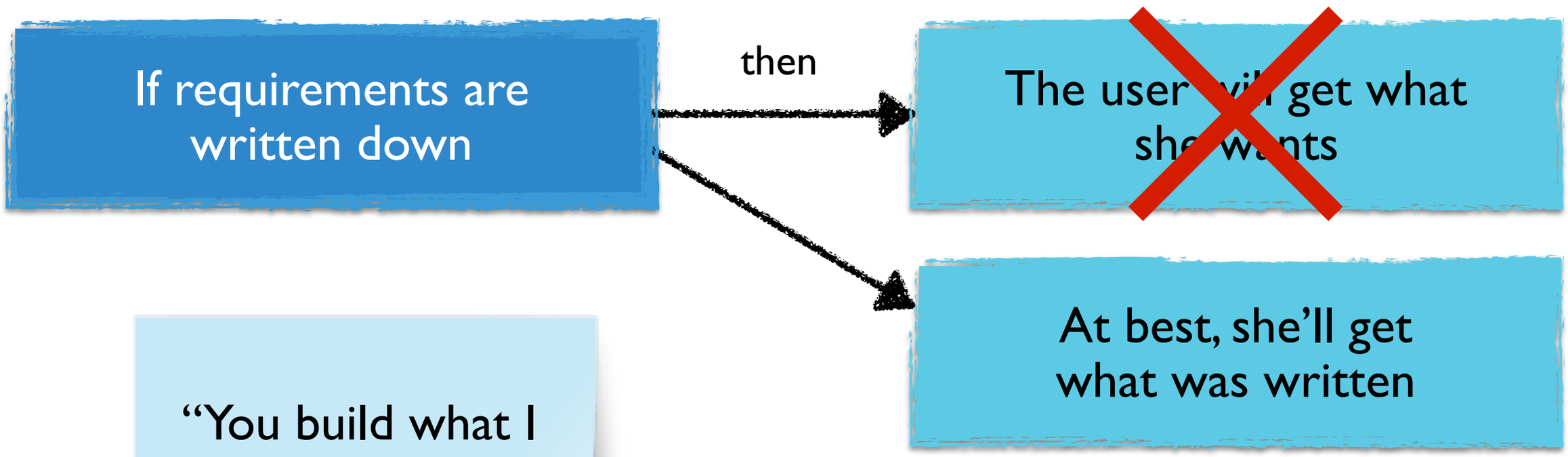
As a frequent flyer, I want to request an upgrade.

As a frequent flyer, I want to see if my upgrade cleared.

Agenda

- ☐ What stories are
- ☐ Writing user stories
- ☑ Why user stories

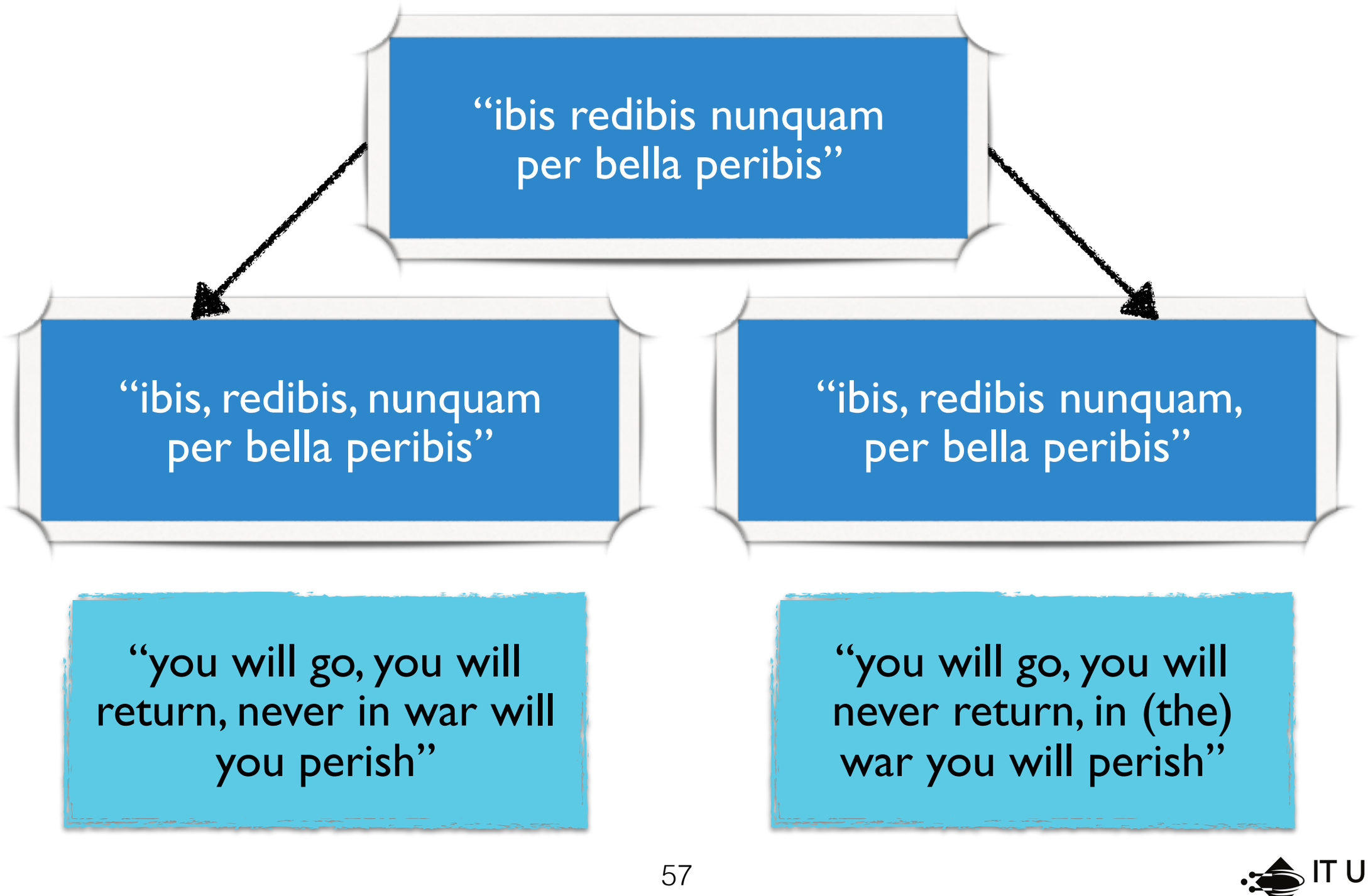
So why user stories?



"You build what I asked for, but it not what I need."

- Shit focus from writing to talking

Words are imprecise



Examples

"Entrée comes with soup or salad and bread"

Which is right?

- (soup or salad) and bread
- soup or (salad and bread)

Examples

The user can enter a name. It can be 127 characters.

- Must the user enter a name?
- Can it be other than 127 chars?

The system should prominently display a warning message whenever the user enters invalid data.

- What does should mean?
- What does prominently display mean?
- Is invalid data defined elsewhere?

Additional reasons

- Stories are understandable
  - Developers and customers understand them
  - People are better able to remember events if they are organised into stories

- Support and encourage iterative development
  - Can easily start with epics and disaggregate closer to development time

Bower, Black, and Turner, 1979. *Scripts in Memory for Text.*

Yet more reasons

- Stories are the right size for planning
- Stories support opportunistic development
  - We design solutions by moving opportunistically between top-down and bottom-up approaches
- Stories support participatory design

Guidon, 1990. *Designing the design process.*

Example: specifications IEEE-style

- The product shall have a steel body
- The product shall have four wheels
- The product shall have a rubber tier mounted to each wheel
- The product shall have a multi speed transmission

Where is your head going?

What if we had stories instead?

As a user, I want to
mow my lawn quickly
and easily.

As a user, I want to
be comfortable while
morning my lawn.

What if we had stories instead?

Most importantly …

Don’t forget the purpose

The story text we write on cards is less important than the conversations we have.

Concluding

Outline

- Literature
  - "The New New Product Development Game"
- Scrum in details
  - Roles
  - Cerimonies
  - Artifacts
- User stories in details
- Next time: more on estimation, velocity, burndown charts, and techniques for forecasting