Project Management

Introduction to Systems Engineering
I2ISE

Introduction

- Why do we need project management?
- Groups vs. teams
- Team roles and activities
- Maintaining a team – group AC
- Project planning and estimation, risk management
- Scrum

Why project management?

- Discussion: Your experiences on project work?

Project teams, roles and activities

Groups vs teams

- What is the difference between a *group* and a *team*?

GROUP	TEAM
Individual accountability	*Individual* and *mutual* accountability
Meet to share information	Meet to discuss, make decisions, solve problems, planning
Focus on *individual* goals	Focus on *team* goals
Produce *individual* work products	Produce *collective* work products
Define *individual* roles, responsibilities, and tasks	Define individual roles, responsibilities, and tasks *to help team do its work*
Concerned with *individual's* outcome and challenges	Concerned with *team* outcome and challenges
Purpose, goals, approach to work shaped by *manager*	Purpose, goals, approach to work shaped by team leader *with team members*

Nobody is perfect – but a team can be

”A group is a matter of balance. Good team-members has strengths and competencies which cover the needs of the group – without doubling strengths and competencies already present. Strengths possessed by some team-members can compensated weaknesses in others. ***Nobody is perfect – but a team can be.****”*

Dr. Meredith Belbin - http://www.belbin.com/

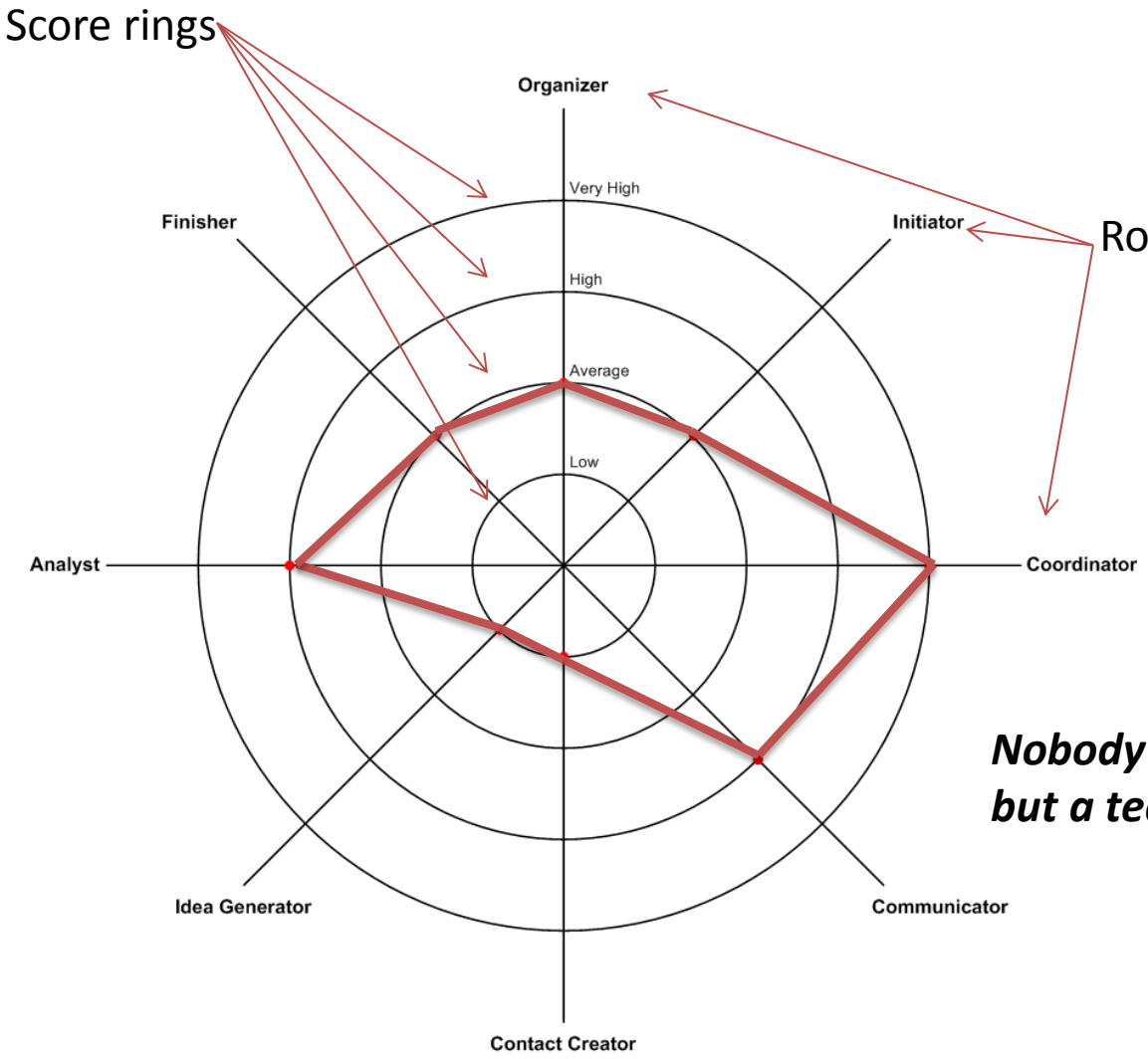
Belbin team roles

- Belbin defines 8 archtypes (*roles*) in a team
- Each role has some positive *qualities* and some allowable *weaknesses*
- The roles of a given team member is identified through a series of *tests*
- The result is a Belbin chart which reveals your preferred roles

Belbin team roles

Type	Positive qualities	Allowable weaknesses
Organiser	Organizing, disciplined, turns ideas into practical actions. Hard working.	Less flexible Skeptical to unproven ideas
Analyst	Sober, strategic. Sees all options. Judges acurately. High intelect.	Lacks drive and ability to inspire others.
Idea generator	Dominating, high intellect. Creative, imaginative, unorthodox.	Ignores routine questions. Too focused on the special problems.
Finisher	Mindful, anxious. Finds errors and omissions. Delivers on time.	Inclined to worry unduly. Reluctant to delegate.
Coordinator	Stable, dominant. Good chairperson, clarifies goals, promotes decision-making, delegates well.	Can be seen as manipulative. Off loads personal work.
Commu-nicator	Stable. Low dominance. Co-operative, mild, perceptive and diplomatic. Listens, averts friction.	Indecisive in crunch situations.
Contact creator	Stable, dominant, enthusiastic, communi-cative, develop contacts.	Over-optimistic. Looses interst once initial enthusiasm has passed.
Initiator	Impatient, dominant, challenging, dynamic, thrive on pressure.	Prone to provocation. Offends peoples feelings.

Belbin team roles - Belbin chart



Nobody is perfect – but a team can be

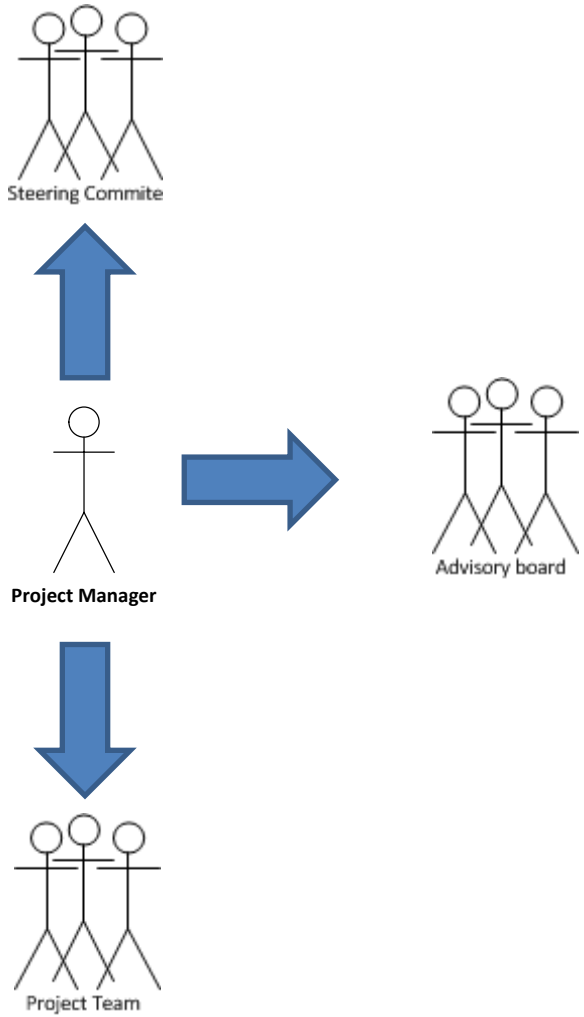
Traditional team roles

- Traditionally, in a team there are some well-known *roles:*
  - Project manager
  - Team members
  - Secretary
- All members assume (at least) one role
- With a role comes *tasks* and *responsibilities*

Team roles – project manager

- What is the project manager's tasks?
  - Manage expectations - internally (team) and externally (stakeholders)
  - Seek information – from team and stakeholders
  - Conduct planning - tasks, plans, manning, preferably with the team
  - Keep information level up - internally and externally
  - Display team culture and behaviour
  - Be the team lightning rod / shield
  - Report to steering comittee
  - …

Project manager - responsibilities



Project manager – personality

- The project manager tasks means that you should select with care
  - Empathic, yet ”cynical”
  - Initiator, yet completor
  - Hghly structured, yet creative
  - Disseminate information, yet shield team
  - …and so forth, and so on.

Team roles – the team members

- What requirements are fair to have to team members?
  - There’s no ”I” in ”team”
  - Responsible
  - Tolerant
  - Loyal to decisions
  - Self-reliant and self-driving
  - Honest
  - Display ”due dilligence”

Teams go through *phases*

Forming	team begins to discuss the task(s) and orientate towards a work plan
Storming	conflicts and tensions emerge - different work styles, expectations, ethics, …
Norming	mutual trust and effective ways of working emerge
Performing	effective work patterns are producing the required results

Maintaining a team - Group AC

- Agreement of Collaboration – a contract!
  - What do you expect of each other? Of the team? Of the result?
  - How will you handle it if someone does not live up to expectations?

- It should contain:
  - The goals and vision of the group
  - The rules for behaviour
  - The rules for collaboration
  - Optionally role assignment dispositions (PM, chairman, secretary)
  - Anything else you feel forms the heart of the group's teamwork

- *Signed by all members!*

Example Group AC

Agreement of Cooperation
Group 4

Vision and goals:
It is our vision to create the world's best, cheapest and most versatile cruise control, and in the process to learn all there is to learn about electronics, C++, …. To do this, we set the following goals:
…

Code of Conduct – general:
- *We will respect the opinion of others and strive to discuss in a factual tone*
- *In the event of arising conflicts, we will discuss these openly in the group at group meetings with the aim of handling them early and effectively.*
- *…*

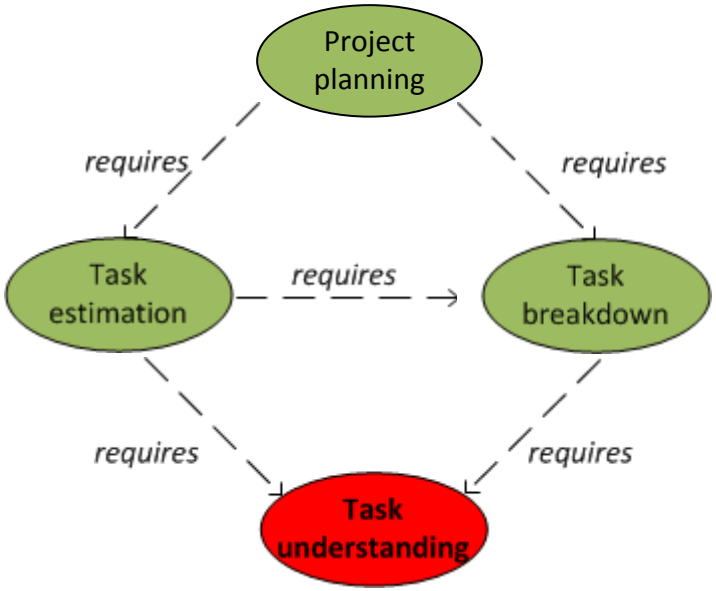
Code of Conduct - meetings:
- *All members of the group participates in the weekly meetings. Absence shall be notified of no later than the morning of the day of the meeting*
- *All members shall participate actively in the meetings*
- *…*

Signed:
(All group members)

Project planning

Project planning

- To have a succesful project, you will need a *plan*



- Thus, to *plan*, we first need to *understand what to plan*!

Planning activities

- Project planning is a *continuous* activity

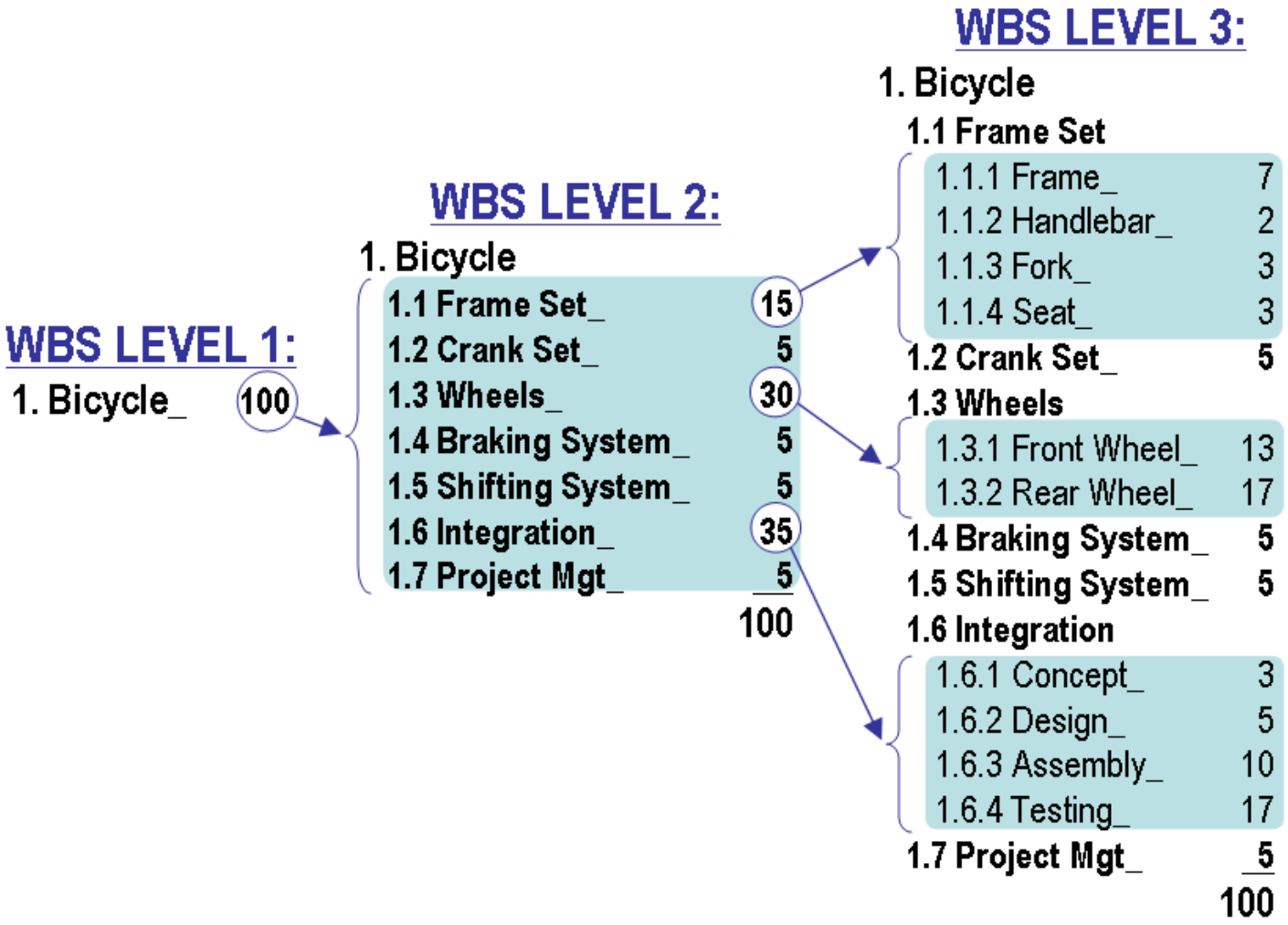
- Initially
  - Break project down into manageable *work packages*
  - Identify *activities* and *milestones*
  - Make *estimates (Estimated Time to Complete* (ETC))
  - Allocate *resources*
  - Create the plan itself

- Continuously
  - Monitor project status and progress
  - Monitor time spent/remaining, compare with milestones
  - Adjust plan/scope of milestones, etc.

Planning – WBS

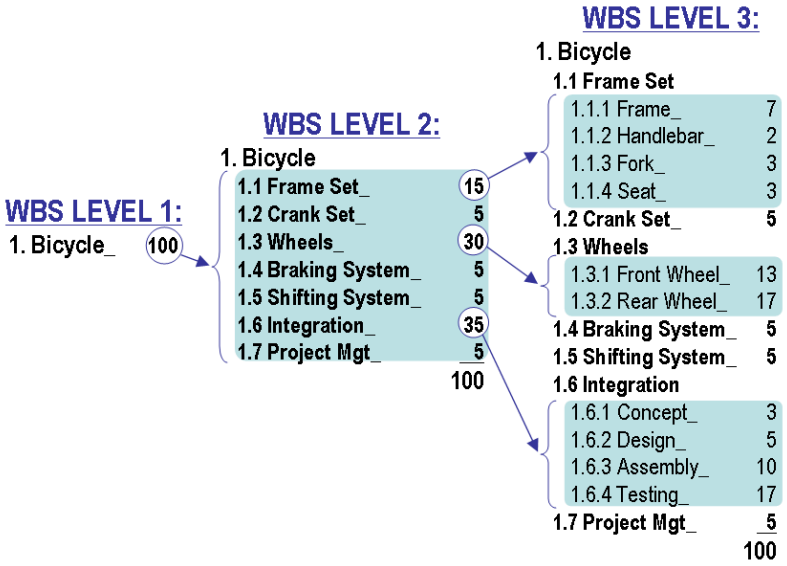
- The work in a project can be broken down in a *Work Breakdown Structure* (WBS)
  - A tree structure containing ever-finer divisions of work
- The WBS leaves should be managable, well-defined, ”estimatable” pieces of work
  - *Terminal elements* or *work packages (WPs)*
- The WBS is the basis of further planning, e.g. time, cost, manpower, dependencies, …

Planning – WBS



WBS design principles

- 100% rule (recursive)
- *Mutually exclusive* rule
- Plan *outcomes*, not *actions*
  - This does *not* have to be physical products



Your turn!

Introduction to Cartoon Heros (I2ICH1) project:

”Give a description on the three classical cartoon heroes, Superman, Batman, and Spiderman. Compare the three and conclude who would win if they got into a fight”

Your turn!

Exercise 1: Create a WBS for the project

(Think planning, writing, reviewing, etc.)

”Give a description on the three classical cartoon heroes, Superman, Batman, and Spiderman. Compare the three and conclude who would win if they got into a fight”

I2ICH project: Example WBS

WP	Task Name
1	Plan detailed contents
2	Create document template
3	Write contents
4	Create artwork
5	Review and corret report
6	Finish report
7	Hand in report
8	Project Management

I2ICH project: Example WBS

WP	Task Name
1	**Plan detailed contents**
1.1	Introduction
1.2	**Presentation of characters**
1.2.1	Spiderman section
1.2.2	Superman section
1.2.3	Batman section
1.3	Comparison
1.4	Conclusion
2	**Create document template**
3	**Write contents**
3.1	Introduction
3.2	**Presentation of characters**
3.2.1	Spiderman section
3.2.2	Superman section
3.2.3	Batman section
3.3	Comparison
3.4	Conclusion
4	**Create artwork**
4.1	Front page
4.2	Spiderman
4.3	Superman
4.4	Batman
5	**Review and corret report**
5.1	Review
5.2	Corrections
6	**Finish report**
6.1	Print contents
6.2	Print front page
6.3	Collect front page and contents
6.4	Bind report
7	**Hand in report**
8	**Project Management**

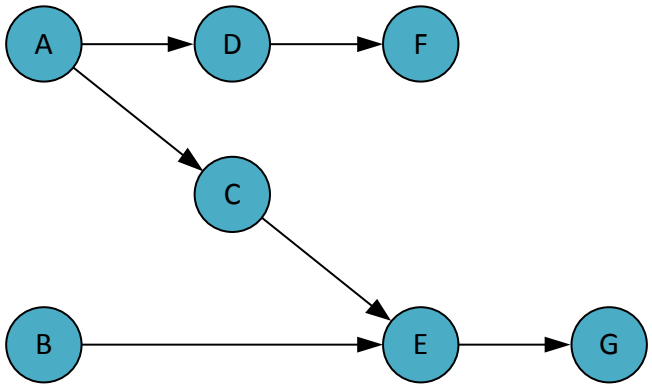
Planning

- Once the project is broken down, you can start to *estimate* and *schedule* your work

- One way to do this:
  - List your WBSs
  - Estimate time to complete (e.g. (P + O + 4N) : 6)
  - Determine dependencies
    - *”C cannot start before A and B is complete…”*

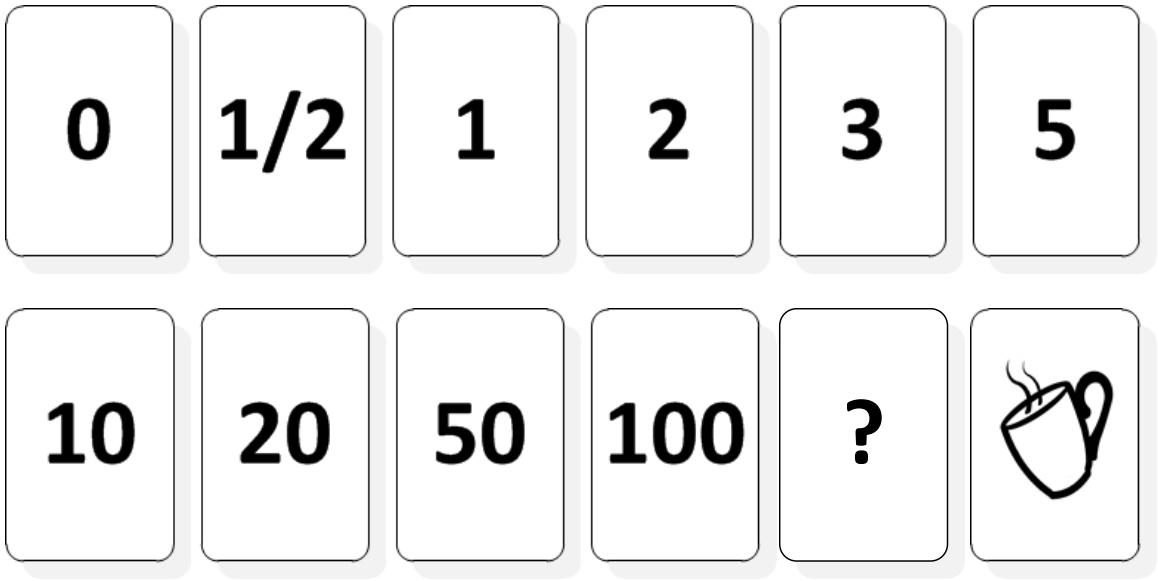
WBS, duration and predecessors

Activity	Predecessor	Time estimates			Expected time
		Opt. (*O*)	Normal (N)	Pess. (*P*)	
A	—	2	4	6	4.00
B	—	3	5	9	5.33
C	*A*	4	5	7	5.17
D	*A*	4	6	10	6.33
E	*B*, *C*	4	5	7	5.17
F	*D*	3	4	8	4.50
G	*E*	3	5	8	5.17



Planning – planning poker

- Another estimation technique: *Planning poker!* You need:
  - A deck of cards per team member
    - hours, days, or ideal days
    - Question mark (*cannot estimate – defer*)
    - Coffee cup (*I need a break!)*
  - An egg timer to structure discussion



Planning – planning poker

- *Product owner* (e.g. PM) gives short introduction to tasks contents. *Team members* discuss task – no numbers!
- Each *team member* lays a planning card face down representing his/her estimate of the task
- Everybody calls at the same time
- Team members with low/high estimates offered a *soapbox to* explain estimates.
- Discussion continues
- *Moderator* or *Product Owner* may at any time set the egg timer
- Egg timer rings → discussion stops, new estimate
- Estimation process repeated until consensus is ”reached”

Your turn!

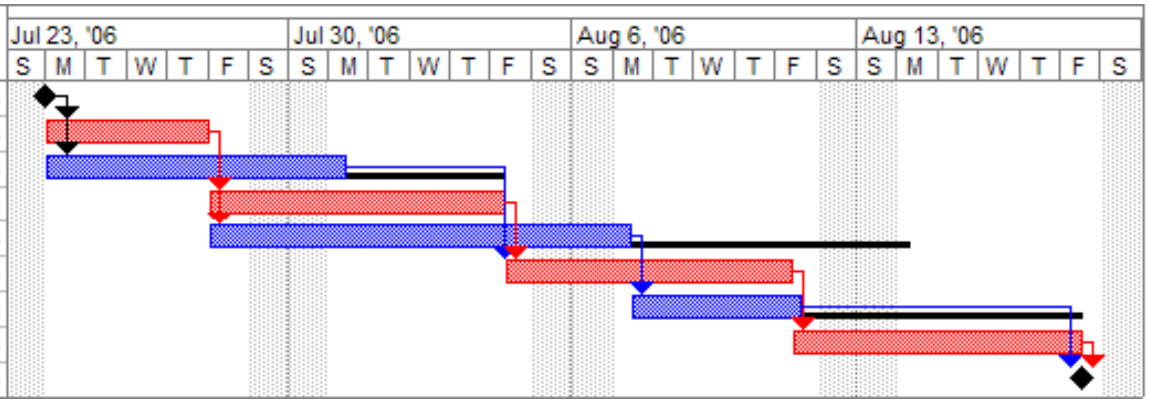
Exercise 2: Give some estimates on durations and determine dependencies for your WBS

”Give a description on the three classical cartoon heroes, Superman, Batman, and Spiderman. Compare the three and conclude who would win if they got into a fight”

Planning – Gantt chart

- With estimates in hand, you can do a *Gantt chart* to show dependencies, duration etc. of tasks
  - Graphical overview
  - Critical paths, milestones, etc.

ID	Task Name	Predecessors	Duration
1	Start		0 days
2	a	1	4 days
3	b	1	5.33 days
4	c	2	5.17 days
5	d	2	6.33 days
6	e	3,4	5.17 days
7	f	5	4.5 days
8	g	6	5.17 days
9	Finish	7,8	0 days



I2ICH project: Gantt chart

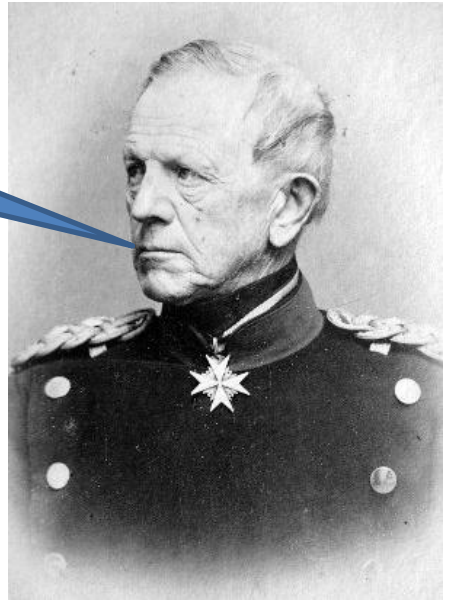
Exercise 3: Use the result of Exercise 2 to create a Gantt chart for your project

”Give a description on the three classical cartoon heroes, Superman, Batman, and Spiderman. Compare the three and conclude who would win if they got into a fight”

Risk Management

- The project plan is objective, but idealized.

No campaign plan survives first contact with the enemy!



Helmuth Karl Bernhard von Moltke
German Field Marshall (1800-1891)

Risk Management

- What can we do to handle risk in the project?

- Risks cannot be avoided, but some can be foreseen and planned for.

- Simple, effective risk mitigation tool:
  1. *Envision* the risk items
  2. *Evaluate* the risk items (probability × consequence)
  3. Make *risk mitigation* / *contingency plan* for each risk item

Risk Management

Description	Prob. 1-5	Conseq. 1-5	Impact 1-25	Risk Mitigation Plan
Members leave team	2	3	6	Mandatory monthly knowledge sharing via team meetings
Subsuppliers delayed	2	5	10	Formal agreement with reimpursement plan
Requirement changes	5	3	15	Frequent demonstrations of product to customer
…	…	…	…	…
…	…	…	…	…

- Extensions:
  - Identify cause
  - Separate *risk mitigation* from *contingency planning*

Scrum

Project Management - Scrum

- Scrum is an iterative, incremental framework for project management

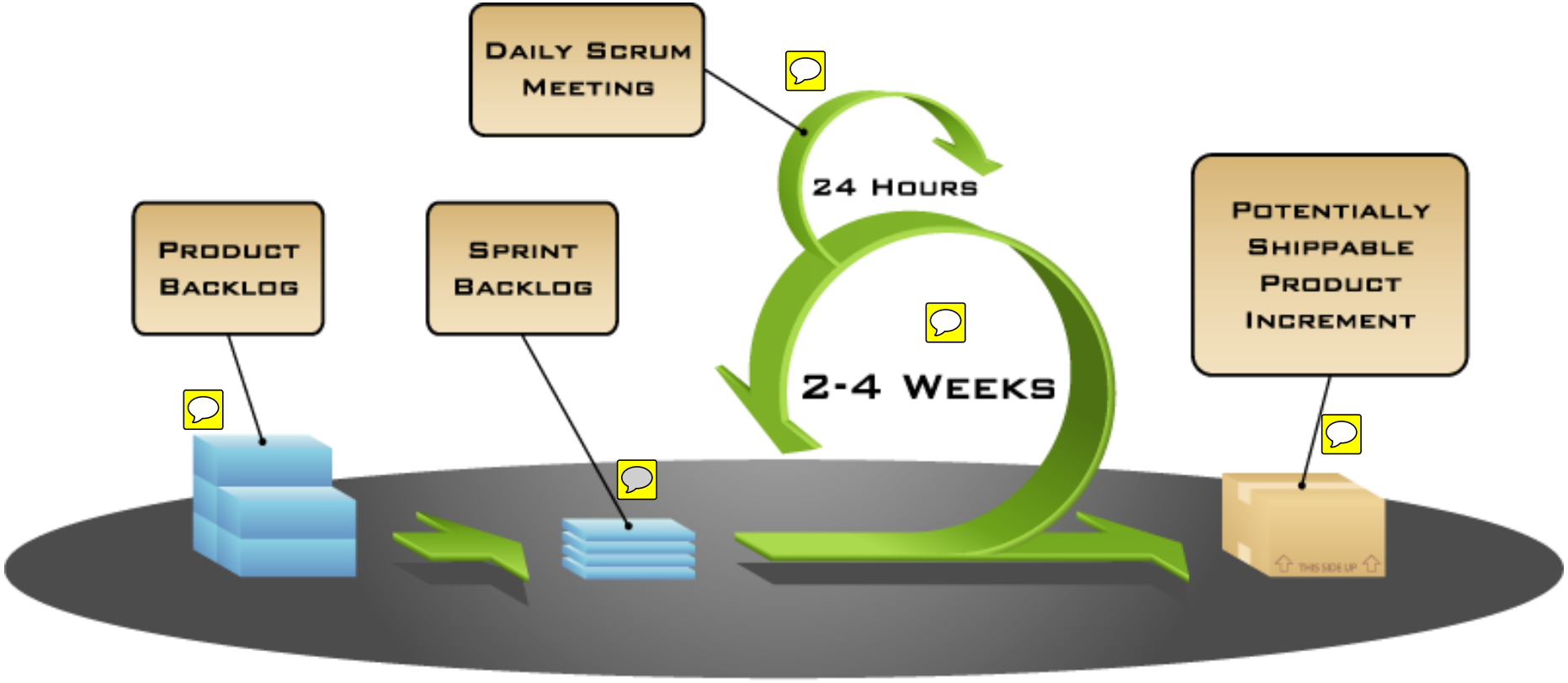
- Involve the whole team in planning, insulate them from changes during *sprints*

- Following slides from www.mountaingoatsoftware.com

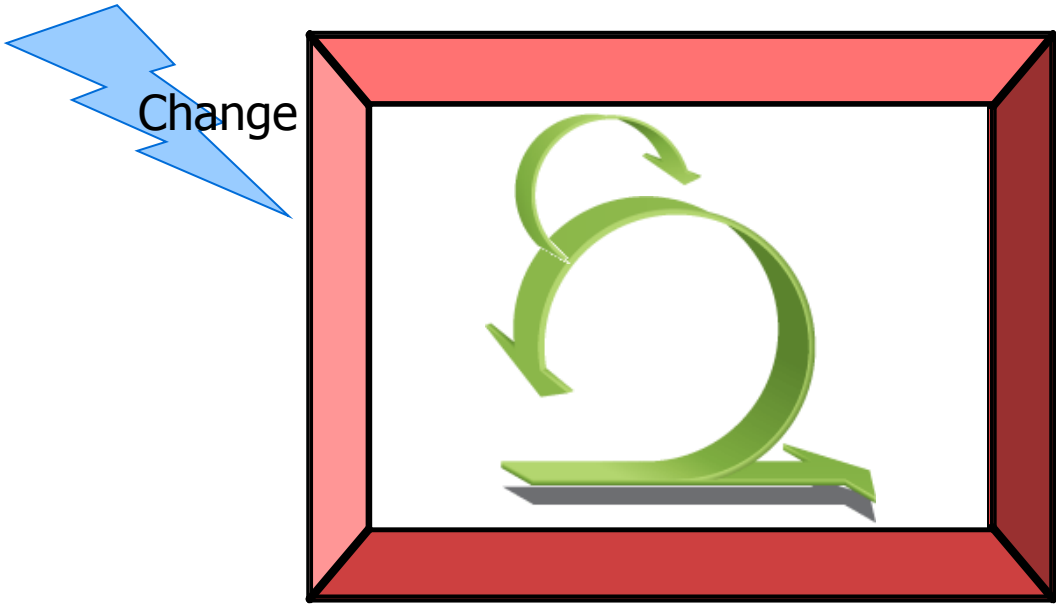
Scrum characteristics

- Self-organizing teams
- Product progresses in a series of 2-4 week “sprints”
- Requirements captured in a “product backlog”
- One of the “agile processes”

Putting it all together

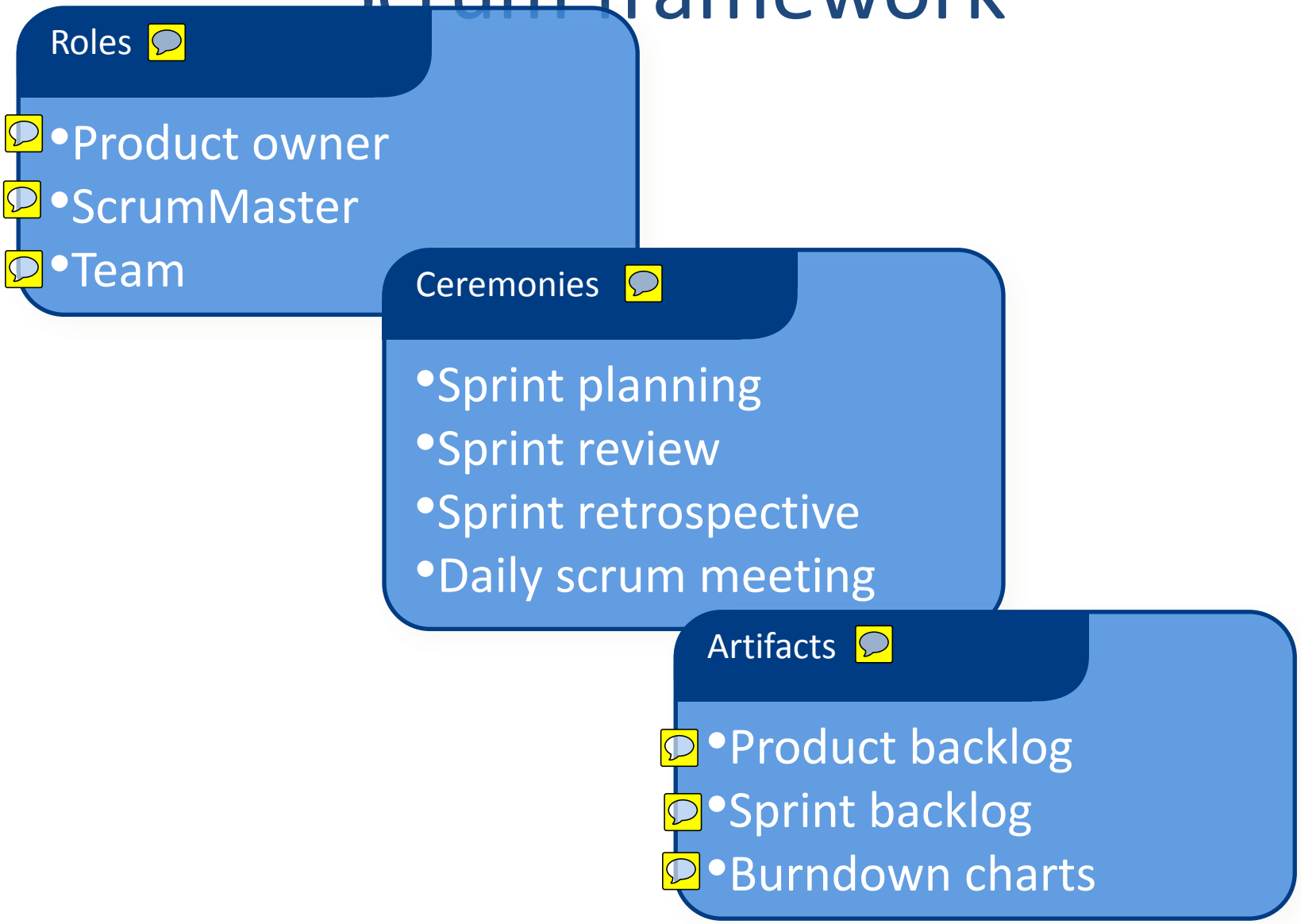


No changes during a sprint



- Sprint durations are planned around how long you can commit to keeping change out of the sprint!

Scrum framework



Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

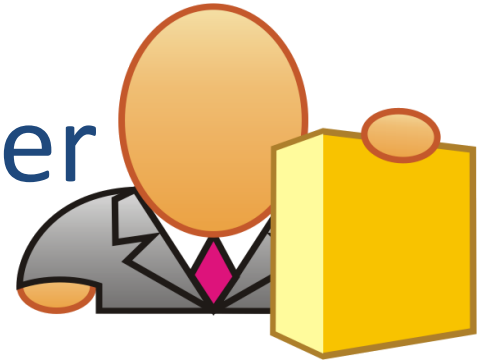
Ceremonies

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

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Roles - Product owner



- Defines the features of the product
- Decides on release dates and content
- Responsible for the profitability of the product
- Prioritizes features
- Adjusts features and priority every iteration
- Accepts or rejects work results
- May “pull the plug” at any time

Roles – Scrum Master

- Represents management to the project
- Responsible for enacting Scrum values and practices
- Removes impediments
- Ensures that the team is fully functional and productive
- Shields the team from external interferences

Roles - The team

- Typically 5-9 people
- Cross-functional:
  - Programmers, testers, user experience designers, etc.
- Members are (ideally) full-time
- Teams are (ideally) self-organizing
- Membership should (ideally) change only *between* sprints

Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

Ceremonies

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

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

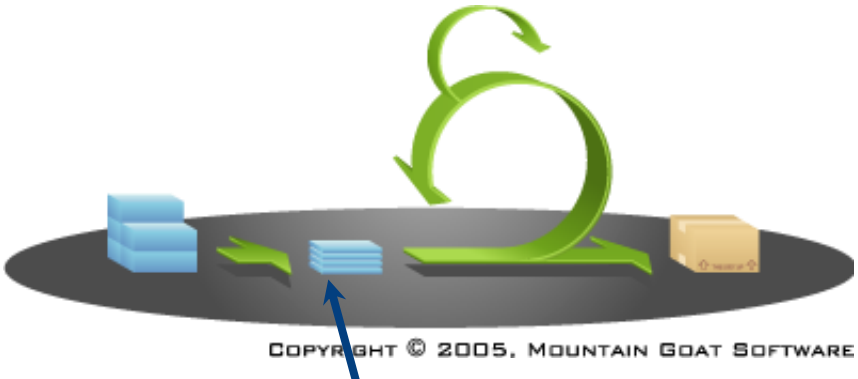
Artifacts - product backlog

- 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

Artifacts - 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

Artifacts - sprint backlog



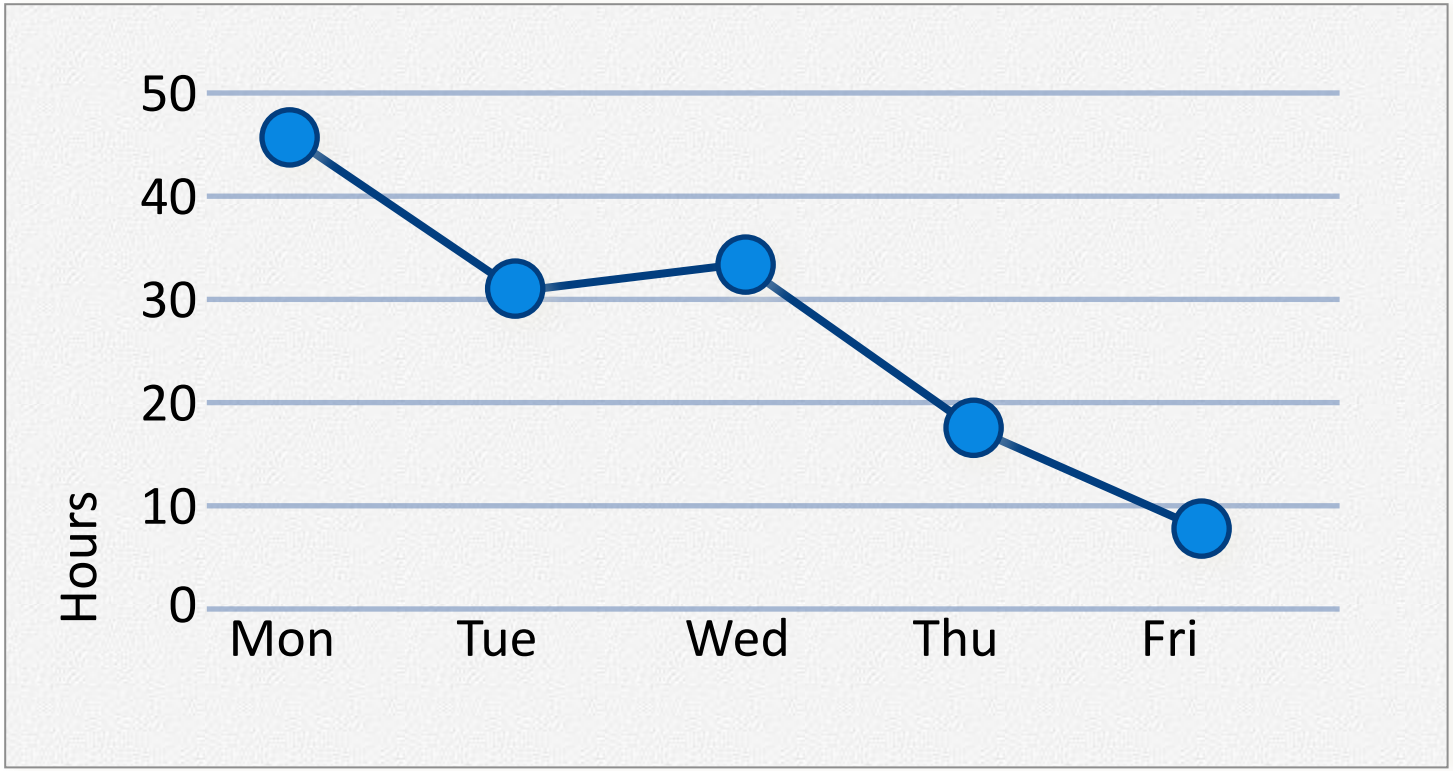
This is the sprint backlog

- The requirements for this sprint, related to *sprint goal*
- Individuals sign up for work of their own choosing
  - Work is never *assigned*
- Estimated work remaining is updated daily
  - Drives sprint *burndown chart*

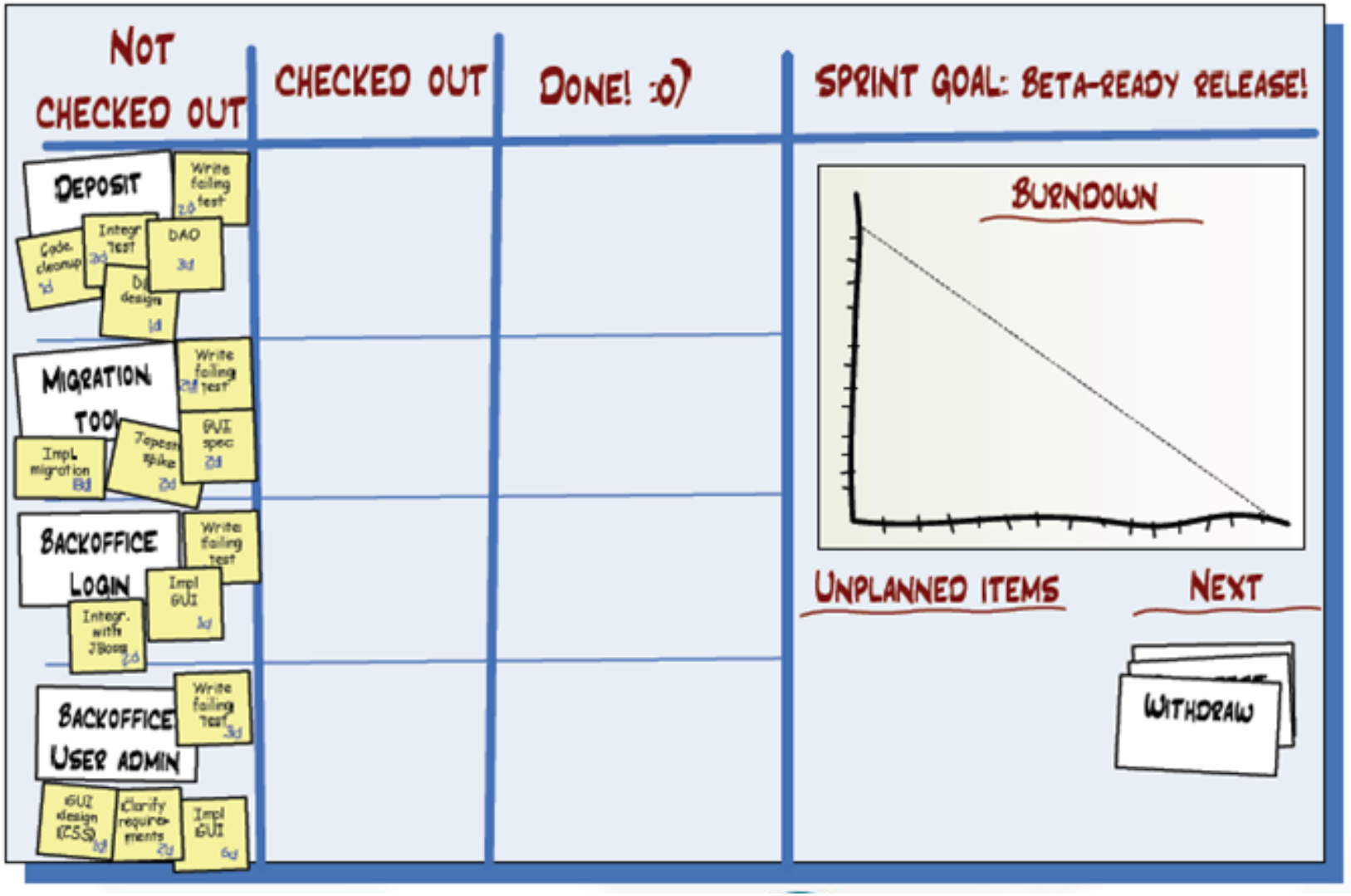
An example sprint backlog

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	

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				



Artifacts – The Scrum Board



Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

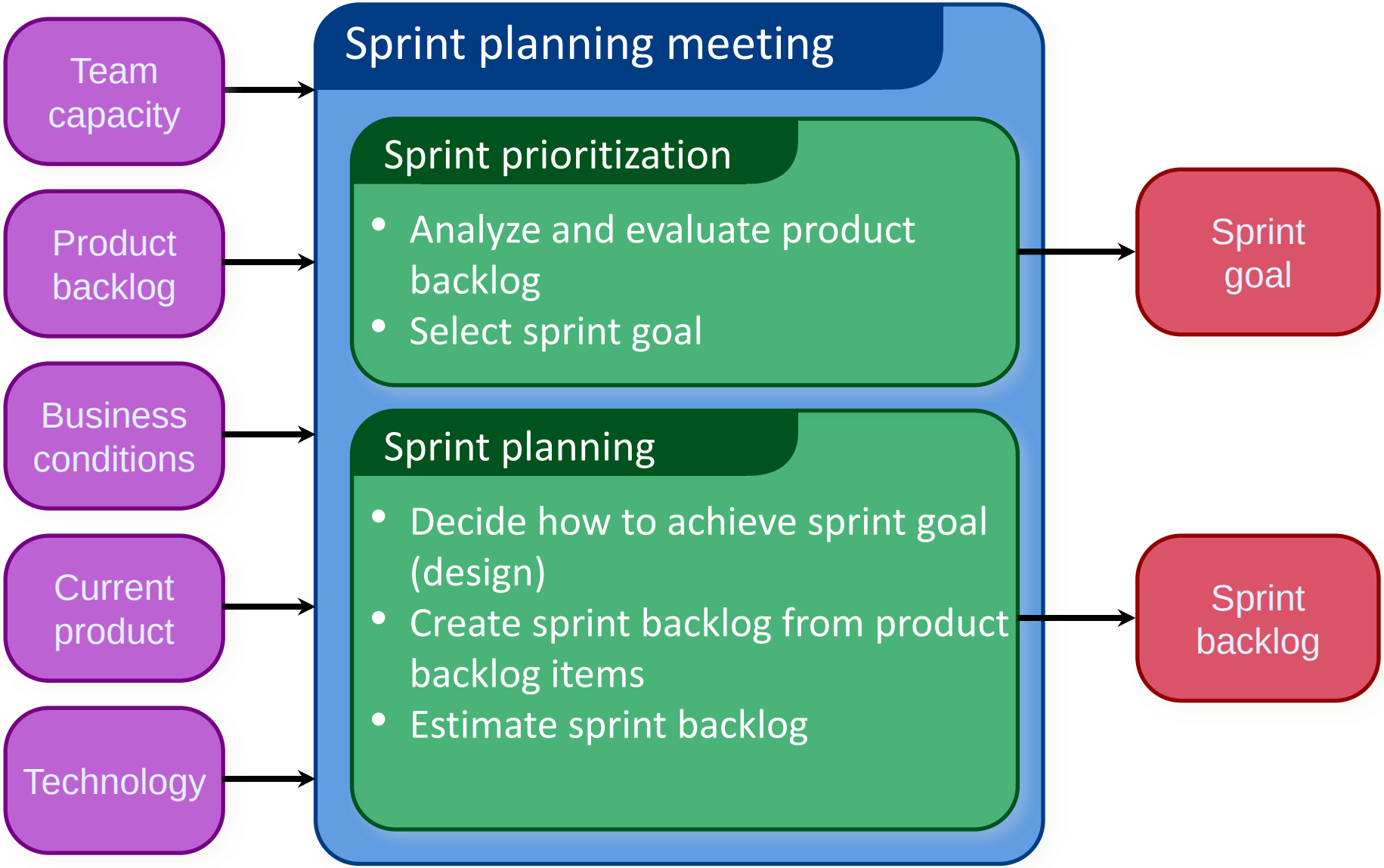
Ceremonies

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

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Ceremonies - Sprint planning



Ceremonies - Sprint planning

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

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

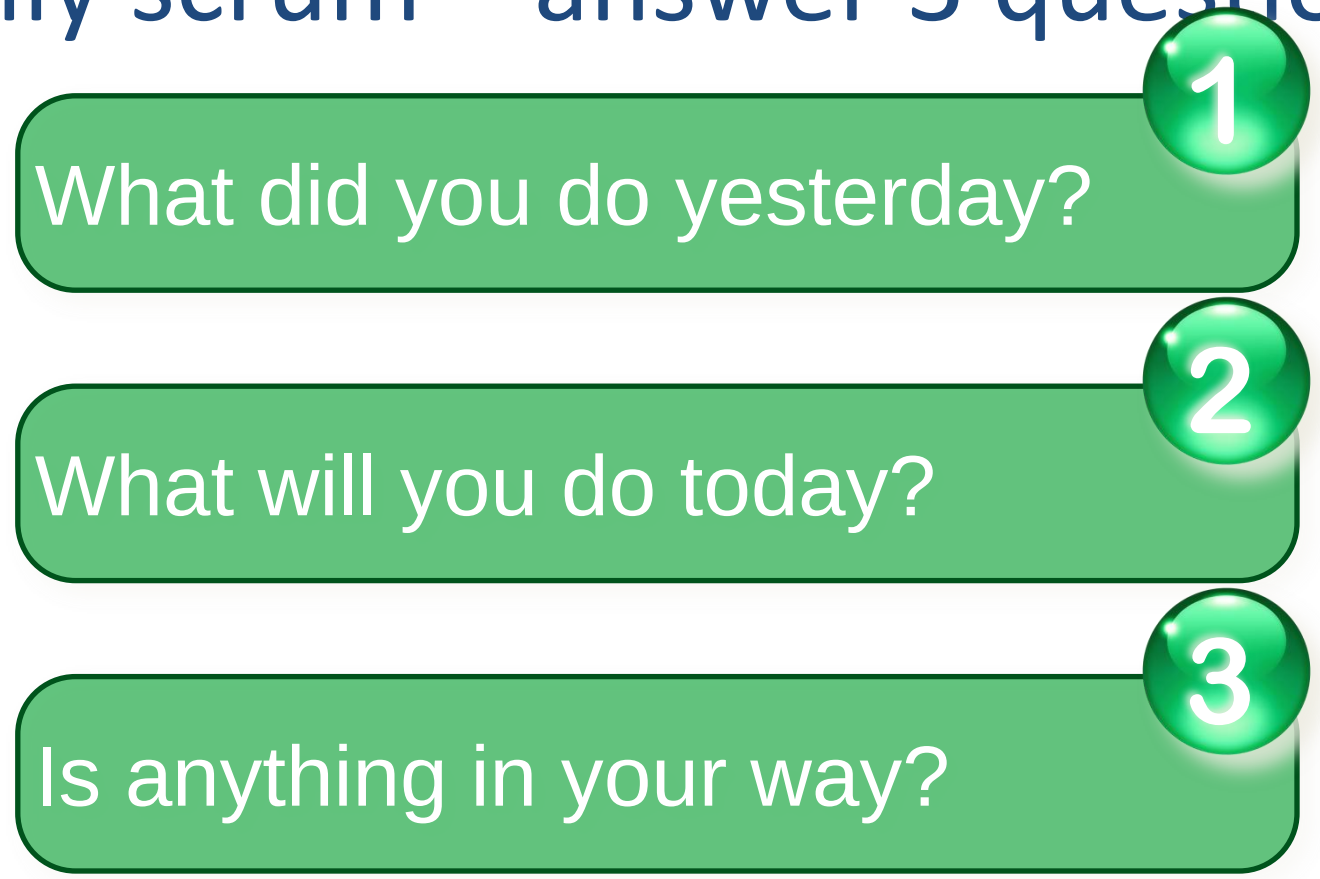
Code the middle tier (8 hours)
Code the user interface (4)
Write test fixtures (4)
Code the foo class (6)
Update performance tests (4)

Ceremonies - daily scrum

- 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

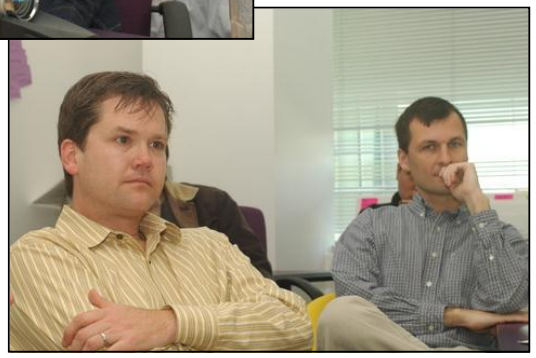
Daily scrum – answer 3 questions



- These are *not* status for the ScrumMaster
  - They are commitments in front of peers

Ceremonies - The sprint review

- Team presents what it accomplished during the sprint
- Typically a demo of new features or underlying architecture
- Informal
- Whole team participates
- Invite the world



Ceremonies– Sprint retrospective

- Periodically take a look at what is and is not working
- Typically 15–30 minutes
- Done after every sprint
- Whole team participates

Sprint retrospective - Start / Stop / Continue

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

Start doing

Stop doing

Continue doing