DEPARTMENT OF COMPUTER & INFORMATION SCIENCES

University of Strathclyde Science

Software Engineering

Lecture 2: Project Management

Billy Wallace
w.wallace@strath.ac.uk

CS993

Lecture Outline

- Project management.
- Agile methods and Scrum.

What is project management?

- Wikipedia says: project management is the practice of initiating, planning, executing, controlling, and closing the work of a team to achieve specific goals and meet specific success criteria at the specified time.
- Projects are time limited.
- Management is a system of processes and controls required to achieve the strategic objectives set by the organisation.
- PM is not “business as usual”, so technically it doesn’t cover maintenance of software.

Project Management Body of Knowledge

- Much like SWEBOK that I referenced in the first lecture, there is also a PMBOK - which is unfortunately no longer a free download. Wikipedia has a nice outline though along with a link to a summary document.
- This has a software extension that talks specifically about how to manage software projects.
- Basically project management is as big a field as software engineering, so don't expect to cover it all today.

What's different about software development?

- Clients often don't know what is needed or what is feasible.
- Clients often lack appreciation for the complexities inherent in software engineering, particularly regarding the impact of changing requirements.
- It is likely that increased understanding and changing conditions will generate new or changed software requirements.
- As a result of changing requirements, software is often built using an iterative process rather than as a sequence of closed tasks.
- Software engineering necessarily incorporates creativity and discipline. Maintaining an appropriate balance between the two is sometimes difficult.
- The degree of novelty and complexity is often high.
- There is often a rapid rate of change in the underlying technology.

The Peter Principle

- People in a hierarchy tend to rise to their level of incompetence.
- People who are good at programming are often not good at managing people.
- It is important to have a non-technical path where such people can be promoted.
- It is useful to partner “team leads” with project managers instead of trying to find “unicorns” that are good at both functions.

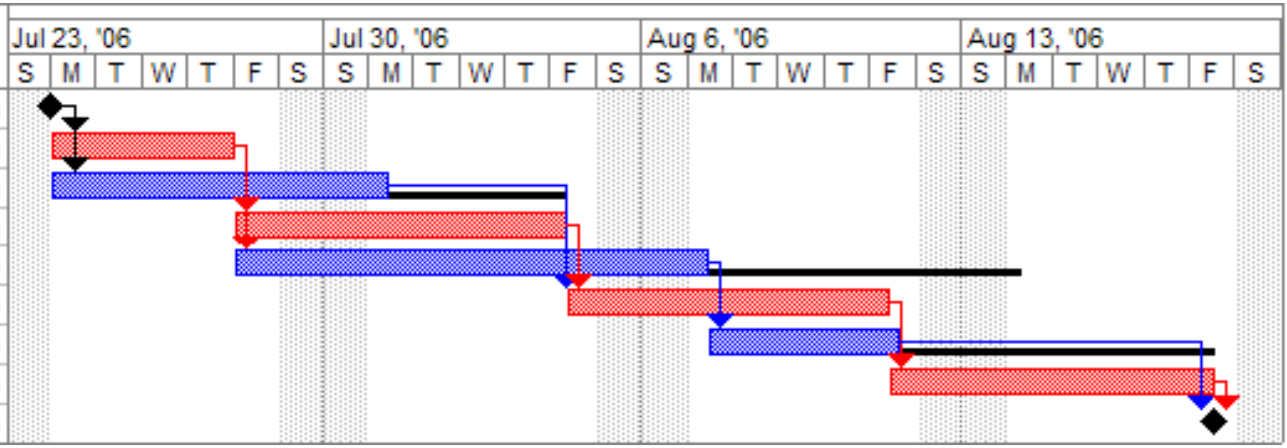
Gantt Charts and Microsoft Project

- There's another chart called a PERT chart that gets discussed for project management, but in reality you will mostly see Gantt charts and these will often be in Microsoft Project.
- This can be overkill for small projects, but a project manager will want to use familiar tools.
- We are interested in knowing where the critical path is.
- It is also useful to be able to know how much "slack" there is in a work package.
- Milestone tasks typically show where deliverables are.

Gantt Charts and Microsoft Project

University of Strathclyde Science

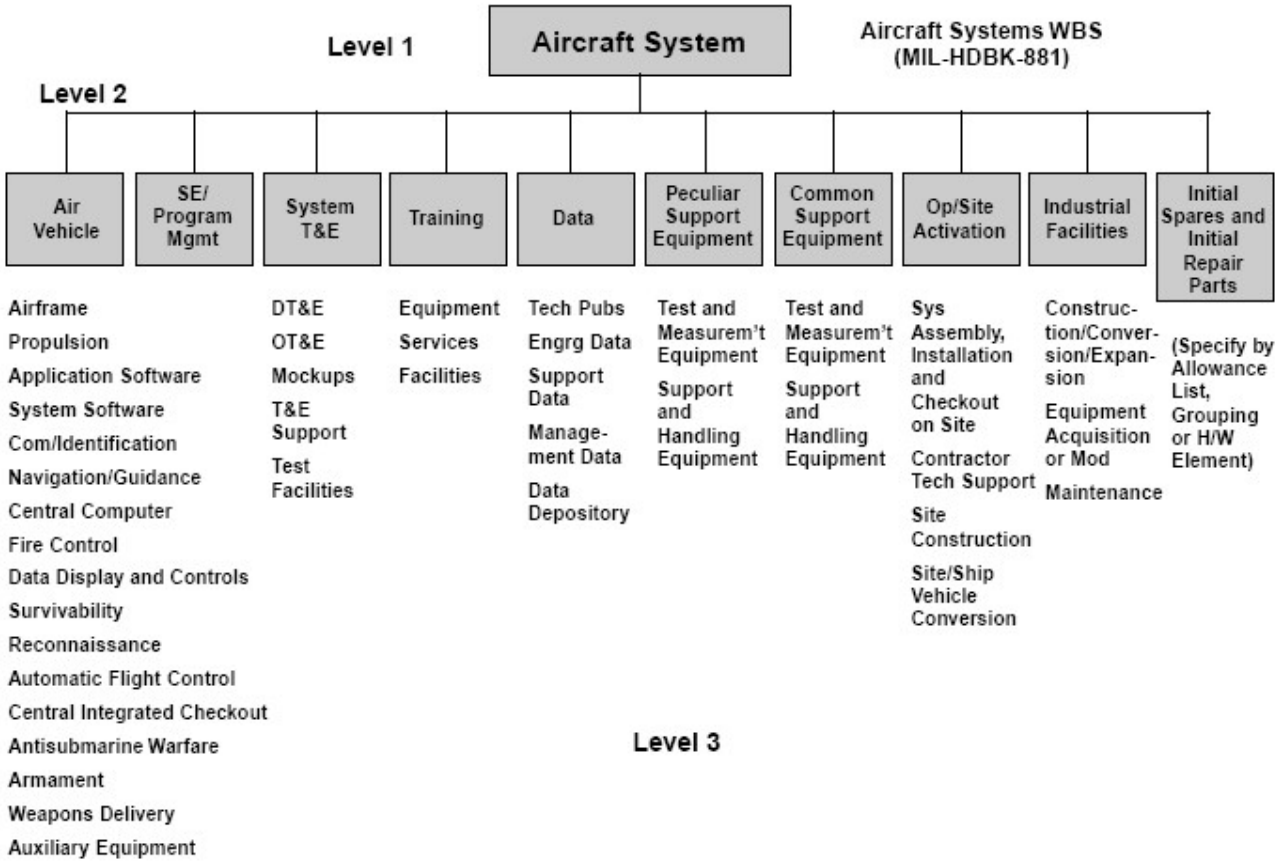
ID	Task Name	Predecessors	Duration
1	Start		0 days
2	a	1	4 days
3	b	1	5.33 days
4	c	2	5.17 days
5	d	2	6.33 days
6	e	3,4	5.17 days
7	f	5	4.5 days
8	g	6	5.17 days
9	Finish	7,8	0 days



Work Breakdown Structures

- Not seen as much, the Work Breakdown Structure (WBS) was developed along with PERT charts.
- WBS is a hierarchical decomposition of a project into phases, deliverables and work packages.
- I like these because they work in the way that software engineers think. We decompose the problem into smaller pieces so that we can work on them, and put them together by abstracting into higher layers.

Work Breakdown Structures



Some Thoughts on Project Management

University of Strathclyde Science

- Think about how this fits with agile methodologies like Scrum.
- Software Engineers sometimes don't respect people that don't have technical skills. They don't realise the value that a good PM can bring to a team.
- PMs can help to navigate the politics of an organisation, keep it at arms-length from you and let you get on with being a techie.
- Techie's are not typically trained in negotiation, so it's good to be able to leave that to someone that understands it.
- Don't confuse management and leadership.
- Even in small software companies, there are a lot more things happening than just software development. We need people to look after the other functions. PMs can get landed with a lot of this in small orgs, e.g. annual development reviews, etc.

University of Strathclyde Science

Manifesto for Agile Software Development

We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on
the right, we value the items on the left more.

University of Strathclyde Science

Agile Methodologies

- Extreme Programming (XP)
- Kanban
- Scrum

University of Strathclyde Science

What is Scrum?

Scrum is a simple yet incredibly powerful set of principles and practices that helps teams deliver products in short cycles, enabling fast feedback, continual improvement, and rapid adaptation to change.
— scrumalliance.org

Recommended reading: The Scrum Guide from the Scrum Alliance website.

University of Strathclyde Science

Scrum Sprints

- Scrum progresses in a series of “Sprints” that typically take 1-3 weeks.
- In each sprint we build, plan, test and review a potentially shippable product.
- We ship the product when it is feature complete.

University of Strathclyde Science

Scrum Roles

- Product Owner
  - Defines the features needed in the product
- Scrum Master
  - “Servant Leader”
- Team
  - Developers, testers, writers, etc

University of Strathclyde Science

Scrum Artifacts

- Product Backlog
  - The product owner creates a prioritised list of user stories.
- Sprint Backlog
  - Highest priority stories go into current sprint backlog along with a size estimate.
- Burndown Chart
  - Shows progress of sprint.

University of Strathclyde Science

Benefits of Burndown Charts

- The burndown chart lets us look at the Sprint Velocity.
- If we keep records of these, we can look at our expected velocity and try to improve our sprint planning.
- If we make a process change, we can look at its impact on our velocity as a way to measure improvements.
- Remember we might add stories to the sprint backlog as we discover some extra work that needs to be done, or if we respond to change.

University of Strathclyde Science

Scrum Ceremonies

- Sprint Planning
  - Everyone meets to discuss user stories and estimate their relative sizes.
  - Output is Sprint Backlog.
- Daily Standup (Scrum)
  - Progress, current work, blockers.
- Sprint Review
  - Demo completed work to product owner and discusses how things can be improved.

University of Strathclyde Science

Useful YouTube Videos

- What is Agile? (Mark Shead)
- Introduction to Scrum - 7 minutes (Uzility)
- Intro to Scrum in Under 10 Minutes (Axosoft)

- Groups should look for tools that can support this.

Summary

- You must plan projects, project management is how we do that.
- We can use structures like Gantt charts and WBSs.
- We are part of a bigger structure and the Project Manager is often our link to that.
- We will focus on Scrum and Agile methodologies.

University of Strathclyde Science

Moving on from here …

- Consider developing software for NASA. Would the approach we've described today be appropriate?
- We want to look at the software engineering tools used to support Agile development.

University of Strathclyde Glasgow