

UiO • Institutt for informatikk

Det matematisk-naturvitenskapelige fakultet

INF5430

**IT Project Management: Overview, traditional
and agile approaches**

Knut H. Rolland



Agenda

- What is a project?
- Overview of the PMI project management standard
- Project planning and monitoring techniques (GANTT, dependencies, critical path, work-breakdown structure, baselines, project triangle)
- Agile project management (burn-down charts, artifacts, meetings)
- Example of a large IT-project

What is a project?

- PMI: "A temporary group activity designed to produce a unique product, service or result".
- Project characteristics
 - has a start and an end
 - has an organization and steering committee
 - has a clear mandate and goals for what to produce – often referred to as 'deliverables'
 - has a defined plan of activities, budget, and schedule

Why organize activities as a project?

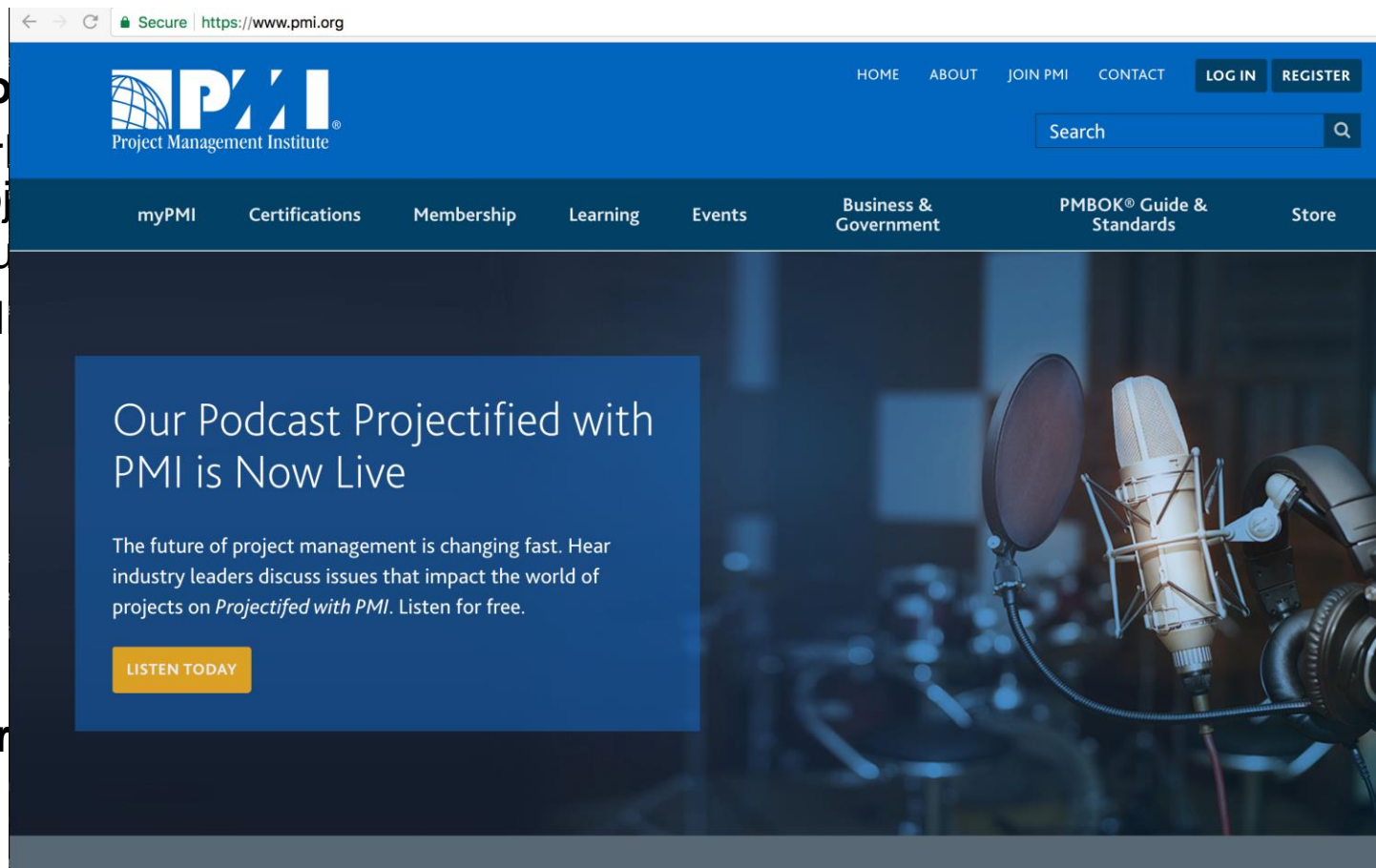
- Solve complex tasks that the line organization is not designed to do
 - Example: develop a complex information system at hospitals
 - Example: develop a case-handling information system at NAV (Norwegian welfare organization)
- Hire a consulting company in order to have access to appropriate knowledge and competence
- Minimize risk for the organization

What is project management?

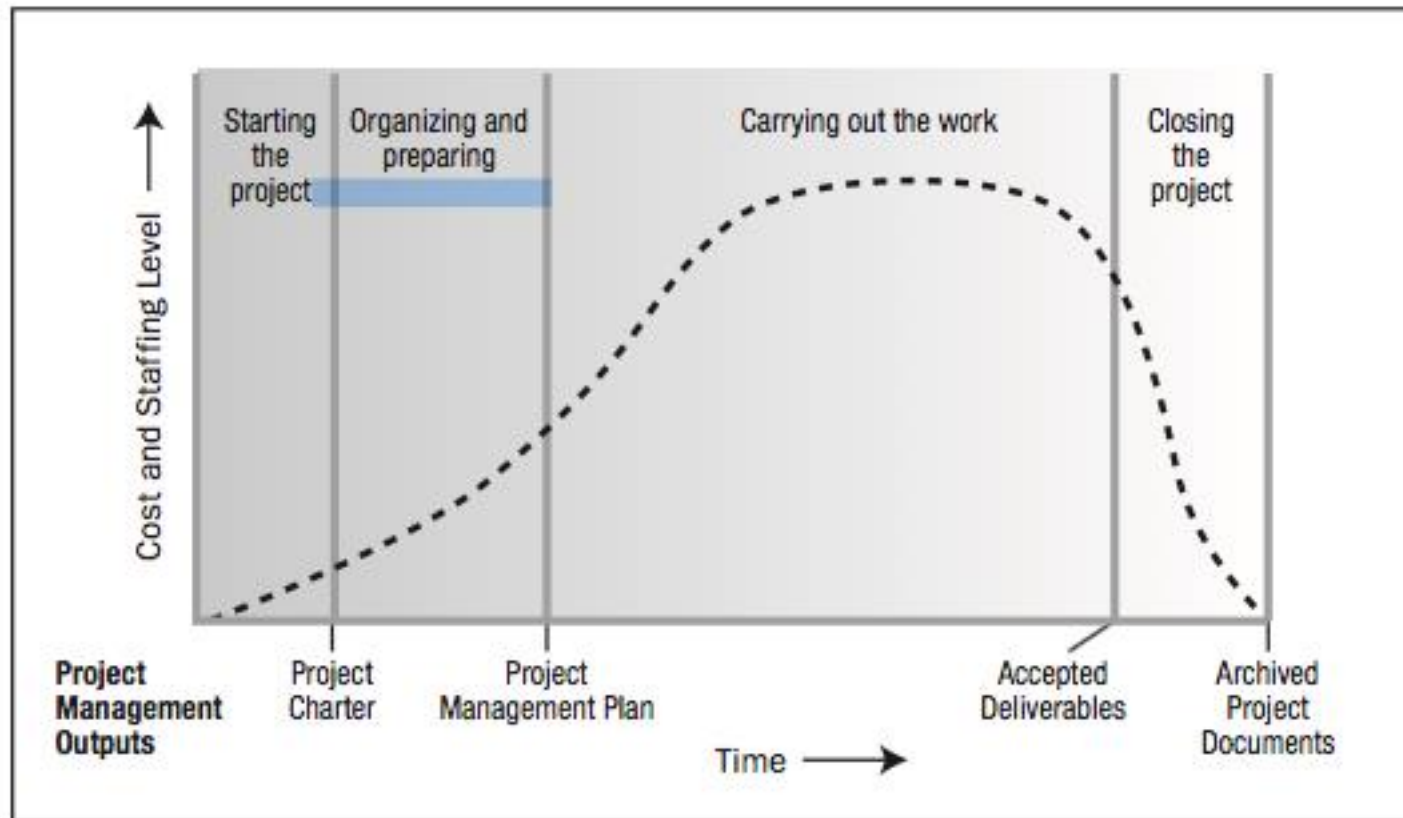
- PMI (def): “The application of knowledge, skills and techniques to execute projects effectively and efficiently.”
- Responsible for managing the project in terms of plans, budgets, resources, competencies, and communication with relevant actors (e.g. steering committee, functional manager)
- **Basic assumption: If correctly planned and executed, a project will achieve its goals.**

The PMI approach

- Pro for- proj Fou
- PM
-
-
- Cer



Project life cycle



PMI is process-oriented

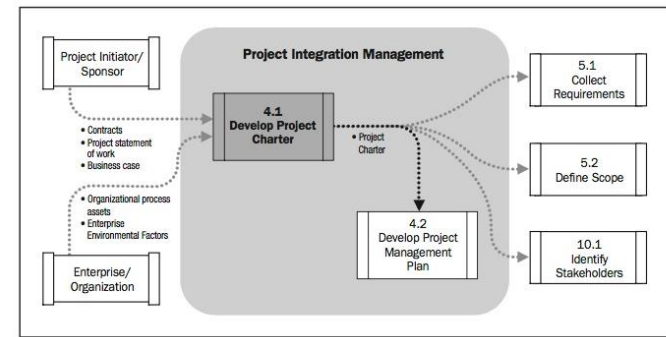


Figure 4-3. Develop Project Charter Data Flow Diagram

Two categories:

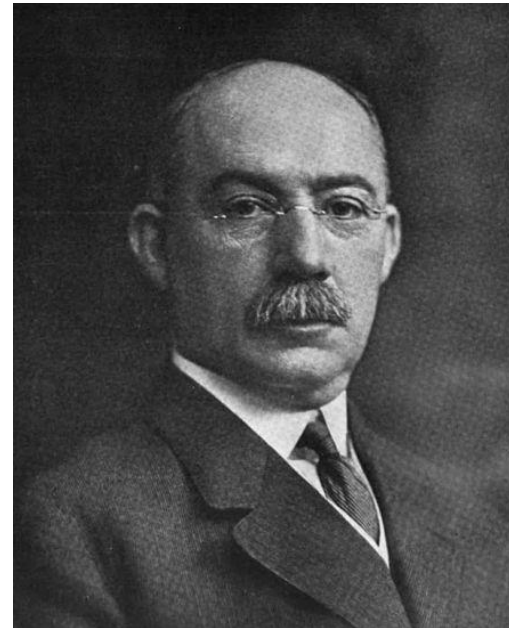
1. **Project management processes** ensure the effective flow of the project throughout its existence.
2. **Product-oriented processes** specify and create the project's product. Product-oriented processes are typically defined by the project life cycle (as discussed in Section 2.1.2) and vary by application area. (PMBOOK Guide p. 37)

The nuts and bolts of project management

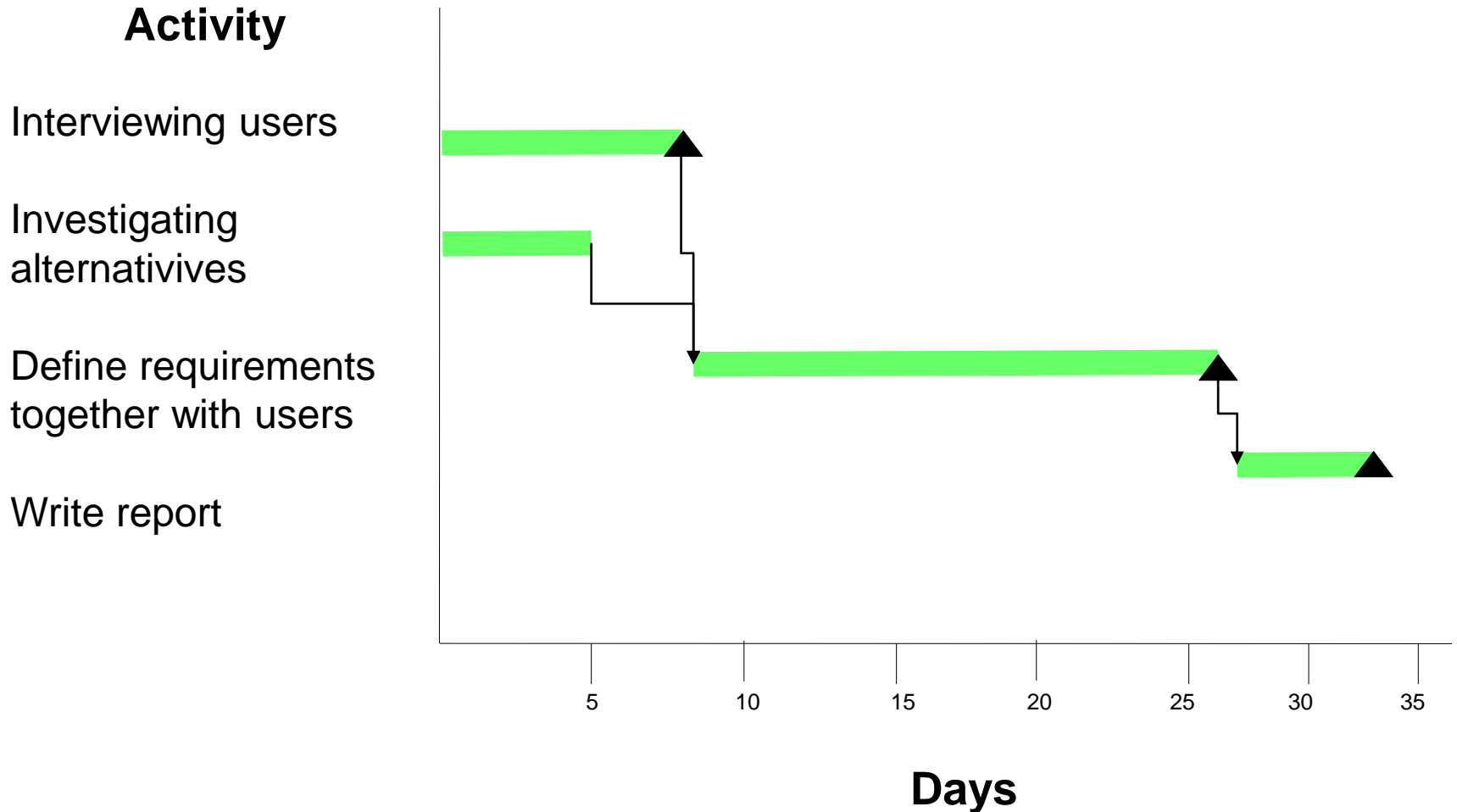
- Project management approaches
- Planning and following up an IT project
- Managing competence, roles and teams in IT projects
- Organizing the project
- Managing the project's stakeholders

Henry Laurence Gantt (1861 –1919)

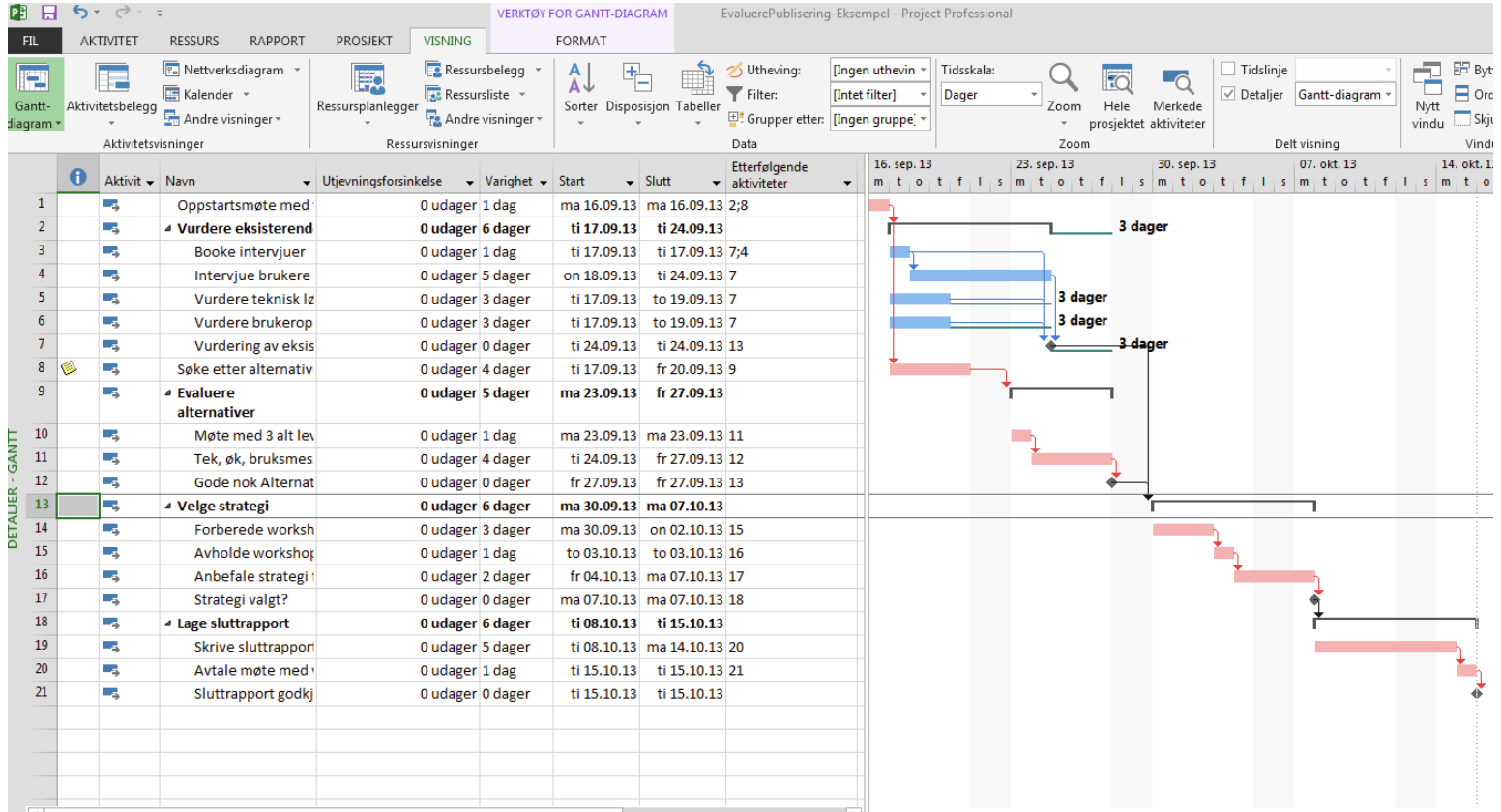
- "Gantt created many different types of charts. He designed his charts so that foremen or other supervisors could quickly know whether production was on schedule, ahead of schedule, or behind schedule."
http://en.wikipedia.org/wiki/Henry_Gantt)



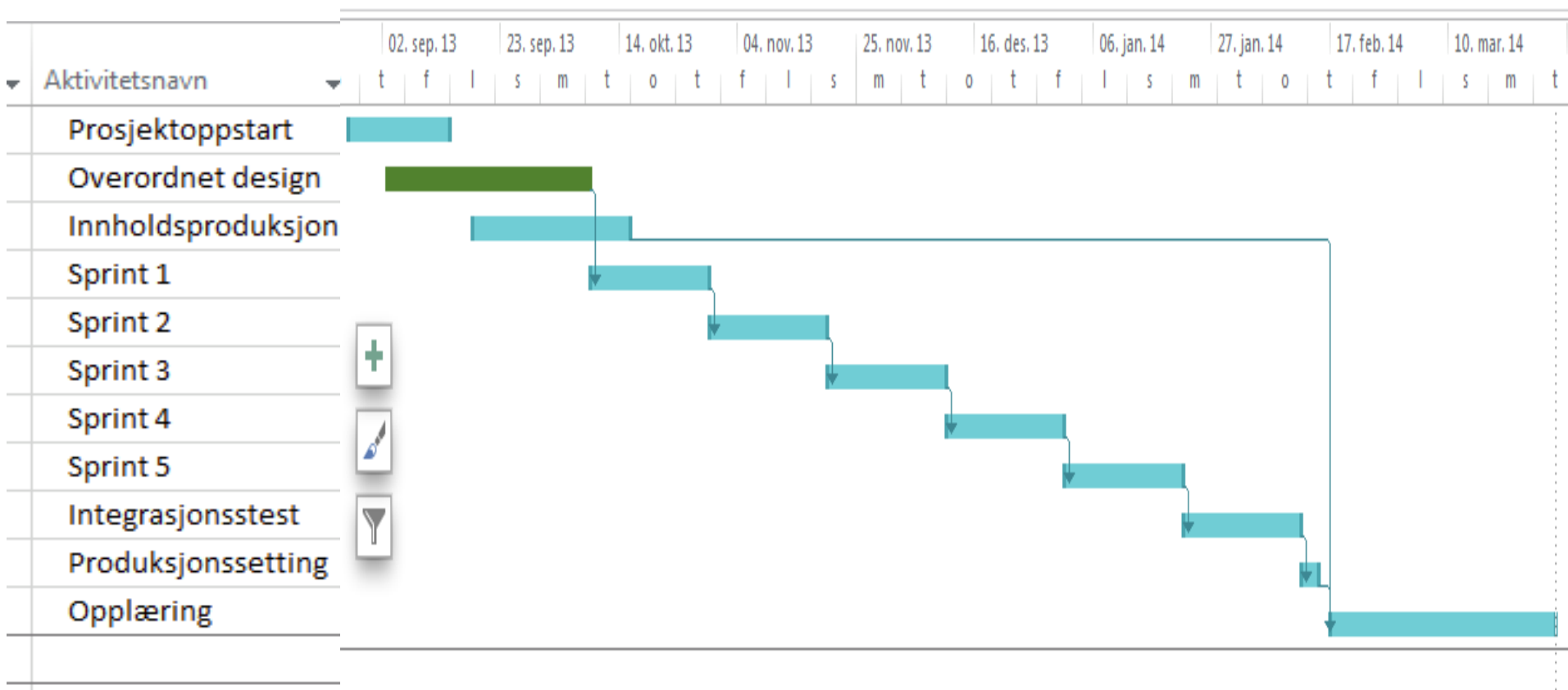
GANTT diagrams



Microsoft Project 2013/6



Example



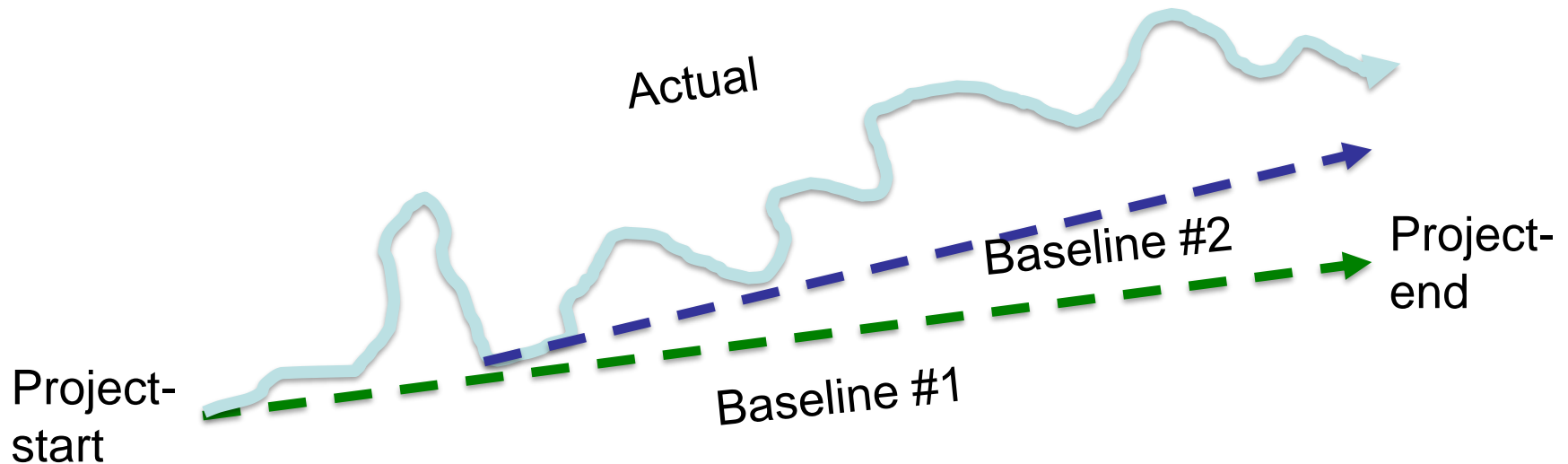
GANTT diagram

Exercise: Develop a GANTT chart for your IN5430 - IT and Management project

- Include:
 - Main activities
 - Dependencies
 - Milestones
- What is the critical path of your project?
- How can you increase the 'slack' of critical activities?

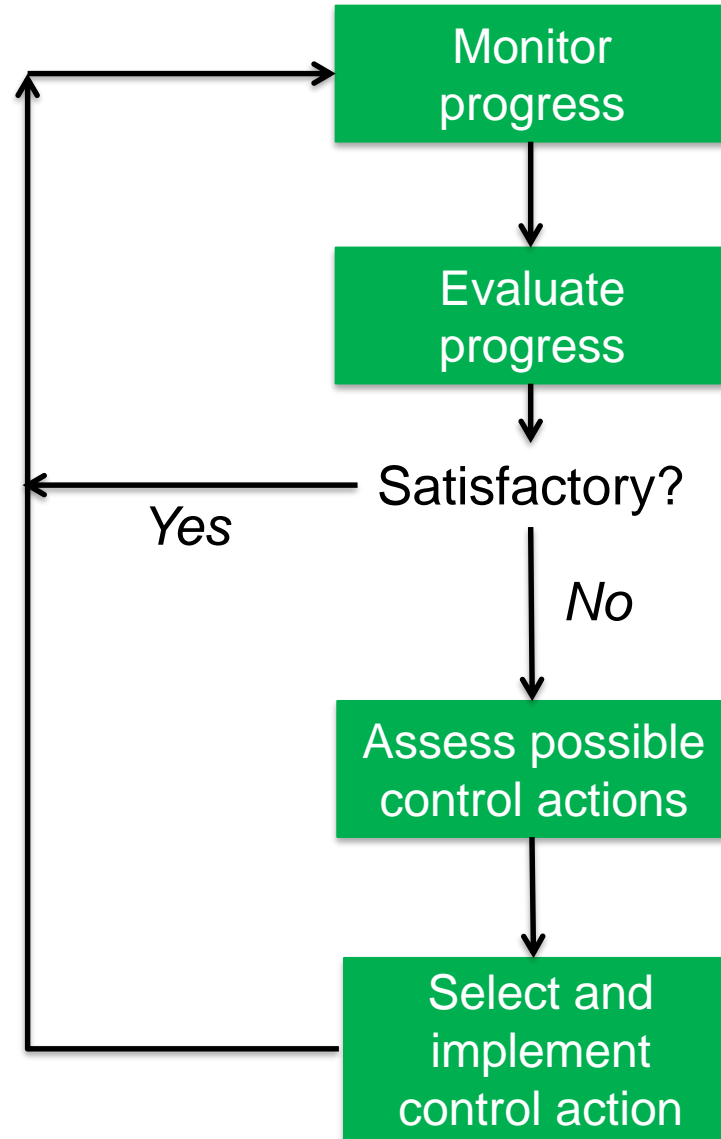
Project baselines

In practice, the project will not exactly follow the plan. If it deviates too much, a new baseline needs to be defined.



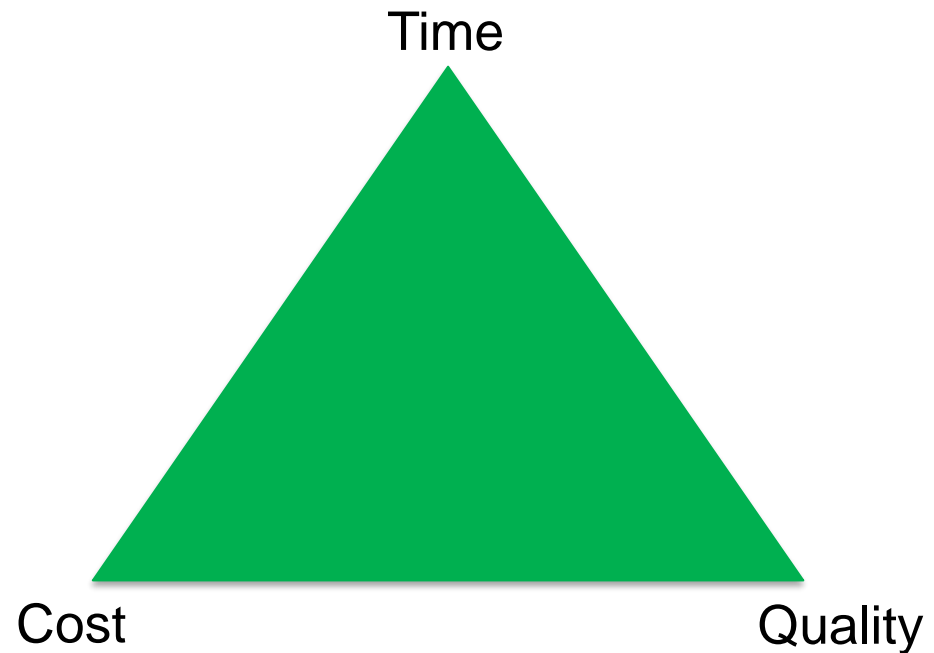
The project manager needs to monitor and control the project in a systematic way (Cadle og Yeates, 2008: s. 208)

Cadle, J., & Yeates, D. (Eds.). (2004). *Project management for information systems*. Pearson education.



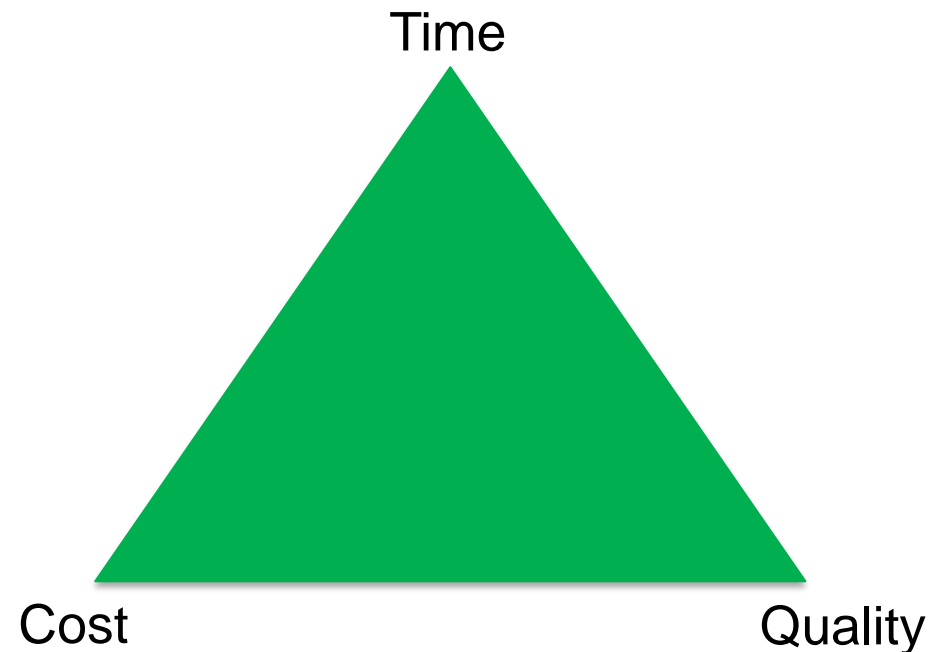
Time/cost/quality triangle

- Project management must balance time, cost and quality.
- Changing one aspect has consequences for the others



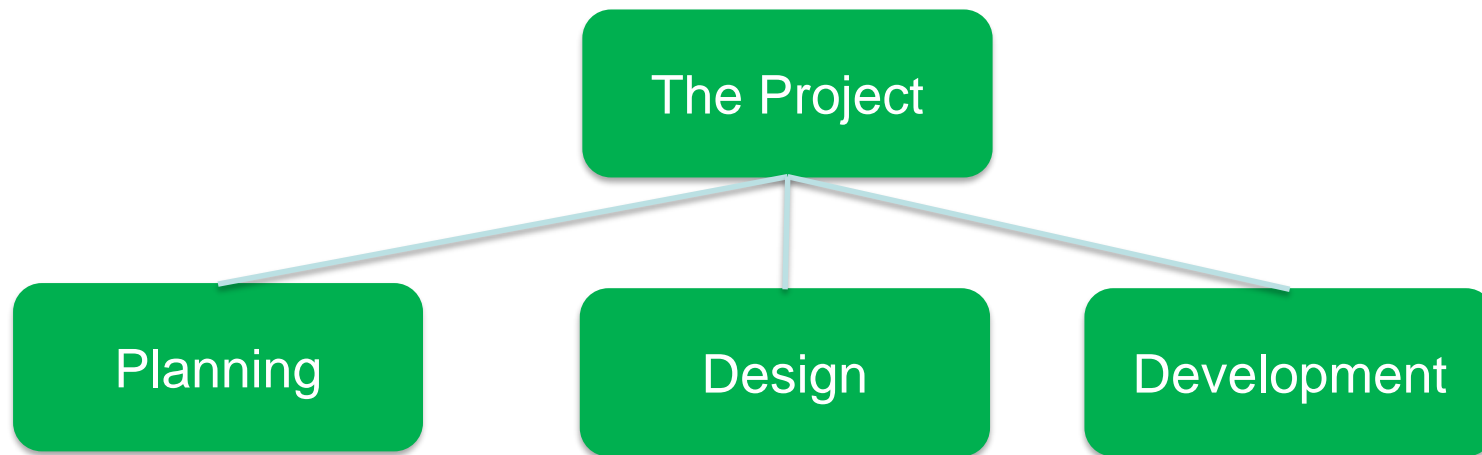
Example: the project has spent too much time (and money) on developing a feature

- The triangle tells you that if you are still going to deliver on time and budget, you have to reduce quality
- Quality can imply both non-functional (e.g. Performance, user experience) and functional requirements (features)



Work breakdown structure

- Dividing the project in smaller parts – activities or work packages
- Presented as a hierarchy of activities with increasing details



Traditional approaches - summary

- Focus on management control and planning
- Detailed plans carried out in a waterfall-like fashion
- Hierarchical organizations often with a top-down approach to project governance
- IT-projects: requirements must be defined up-front and not change too much
- See Spundak (2014)




20% of all IT projects are "Black Swans"

- On average, IT projects in public sector perform remarkably well!
- But, large-scale project especially prone to risks and failures
- Projects that implicates standard software are challenging!!!

Black swans have high cost, schedule and benefit risks, which are hidden in the fat tails of ICT portfolios



Risk comparison, Median, in percent

	Black Swans ¹	Projects with cost overrun	Normal projects	Starved projects
Cost overrun	 +130	+47	+0	-75
Schedule overrun	 +41	+38	+24	n/a
Likelihood	 18	28	31	51

- Even if the median cost overrun is low, risk of cost overruns is high
- **Black Swans mean very high cost and schedule risks**
- And all the projects with a downside risk show significant risk

¹ The statistical expectation value does not converge and is infinite

Budzier, A., & Flyvbjerg, B. (2012). Overspend? Late? Failure? What the data say about IT project risk in the public sector. Commonwealth Governance Handbook, 13, 145-157.

Agile project management approaches

- Co-located and small teams
- Coordination through physical artifacts as well as software tools
- Close interaction with customer
- Self-managed teams

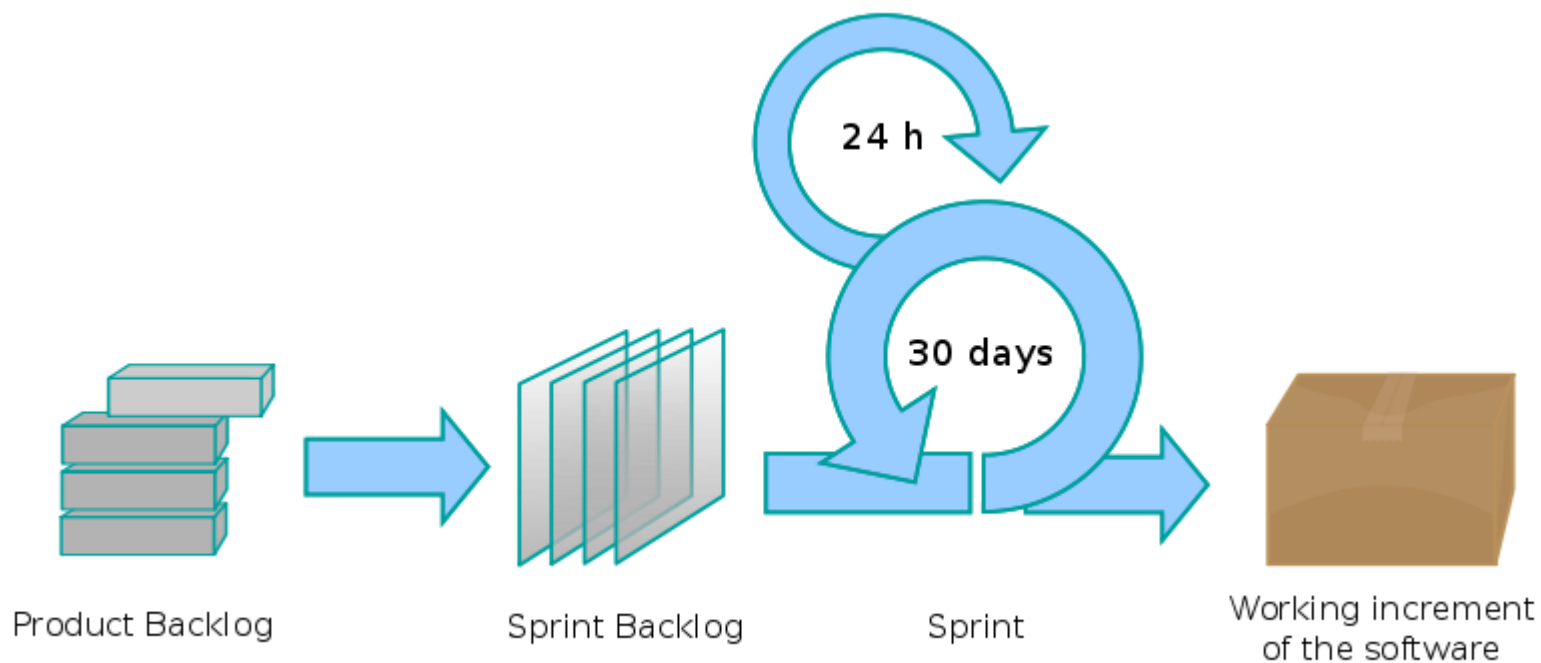
Final Definition of Agility

the continual readiness of an ISD method to rapidly or inherently create change, proactively or reactively embrace change, and learn from change while contributing to perceived customer value (economy, quality, and simplicity), through its collective components and relationships with its environment.

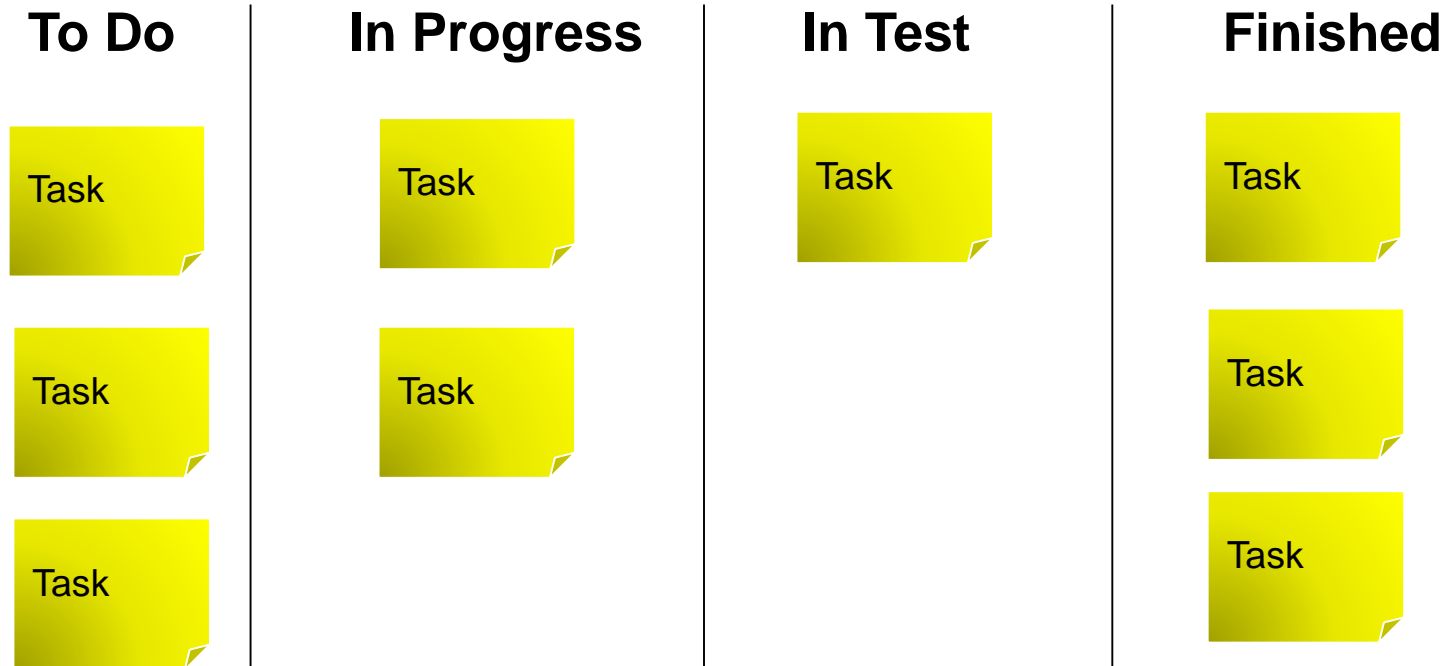
Conboy, K. (2009). Agility from first principles: Reconstructing the concept of agility in information systems development. *Information Systems Research*, 20(3), 329-354.

	Traditional	Agile
Fundamental Assumptions	Systems are fully specifiable, predictable, and can be built through meticulous and extensive planning.	High-quality, adaptive software can be developed by small teams using the principles of continuous design improvement and testing based on rapid feedback and change.
Control	Process centric	People centric
Management Style	Command-and-control	Leadership-and-collaboration
Knowledge Management	Explicit	Tacit
Role Assignment	Individual—favors specialization	Self-organizing teams—encourages role interchangeability
Communication	Formal	Informal
Customer’s Role	Important	Critical
Project Cycle	Guided by tasks or activities	Guided by product features
Development Model	Life cycle model (Waterfall, Spiral, or some variation)	The evolutionary-delivery model
Desired Organizational Form/Structure	Mechanistic (bureaucratic with high formalization)	Organic (flexible and participative encouraging cooperative social action)
Technology	No restriction	Favors object-oriented technology

Agile process: Scrum



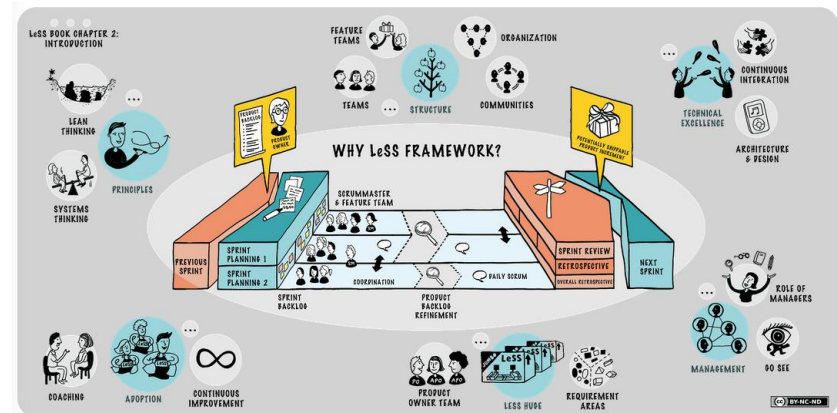
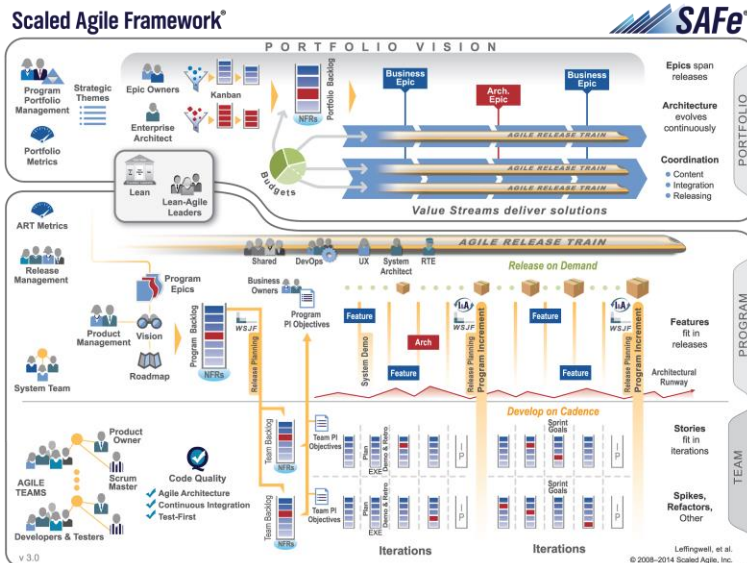
Agile process: Kanban



Agile in larger projects

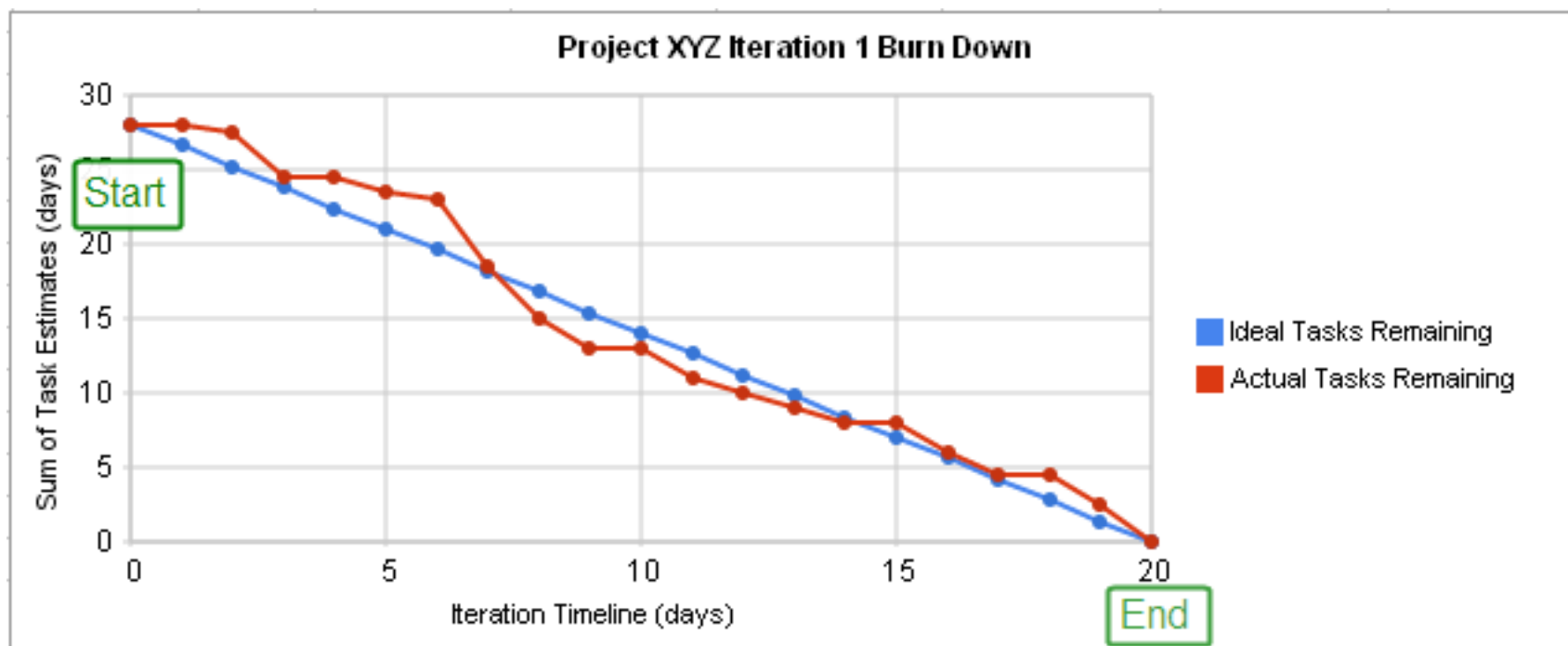


Kniberg, H., *Scrum and XP from the Trenches*: InfoQ, 2007.



www.less.works

Burn-down charts

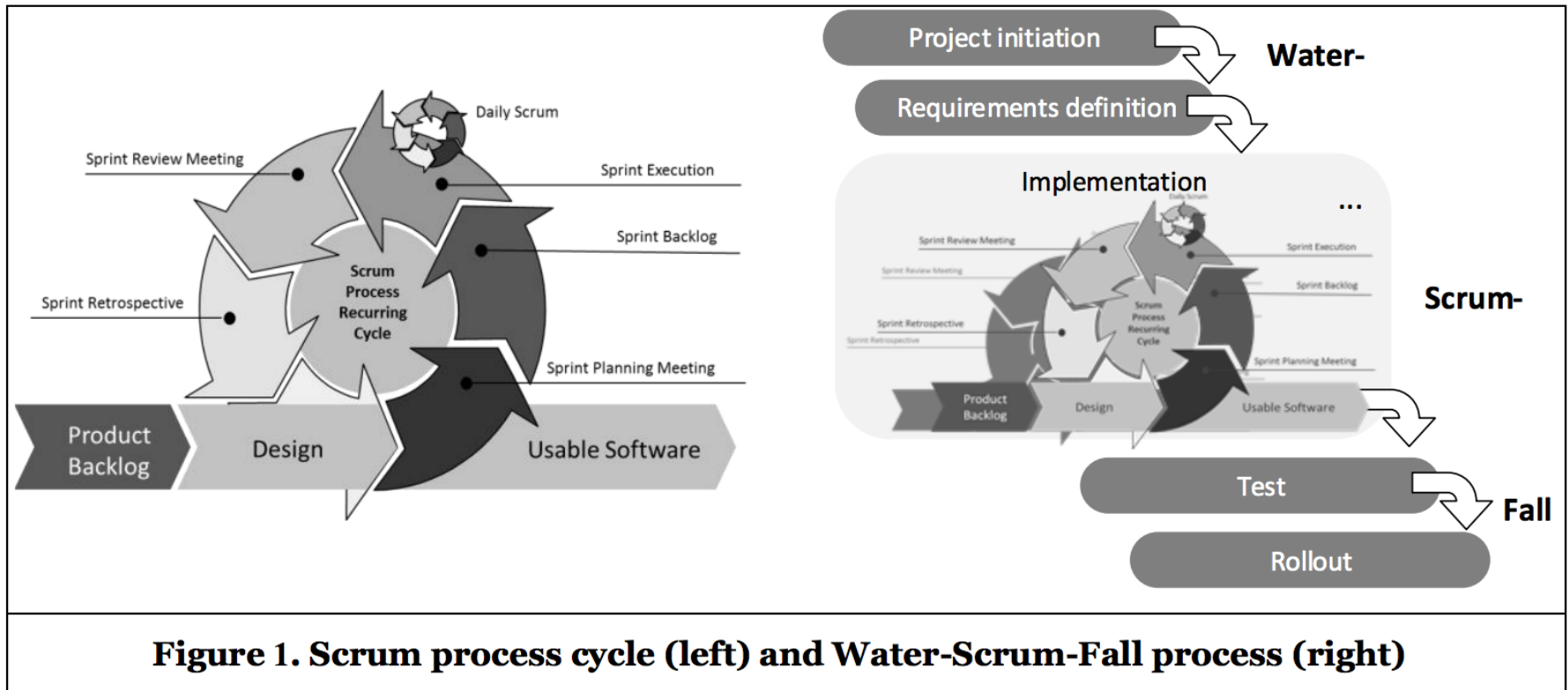


https://en.wikipedia.org/wiki/Burn_down_chart

Extensive use of physical artifacts like whiteboards and notes



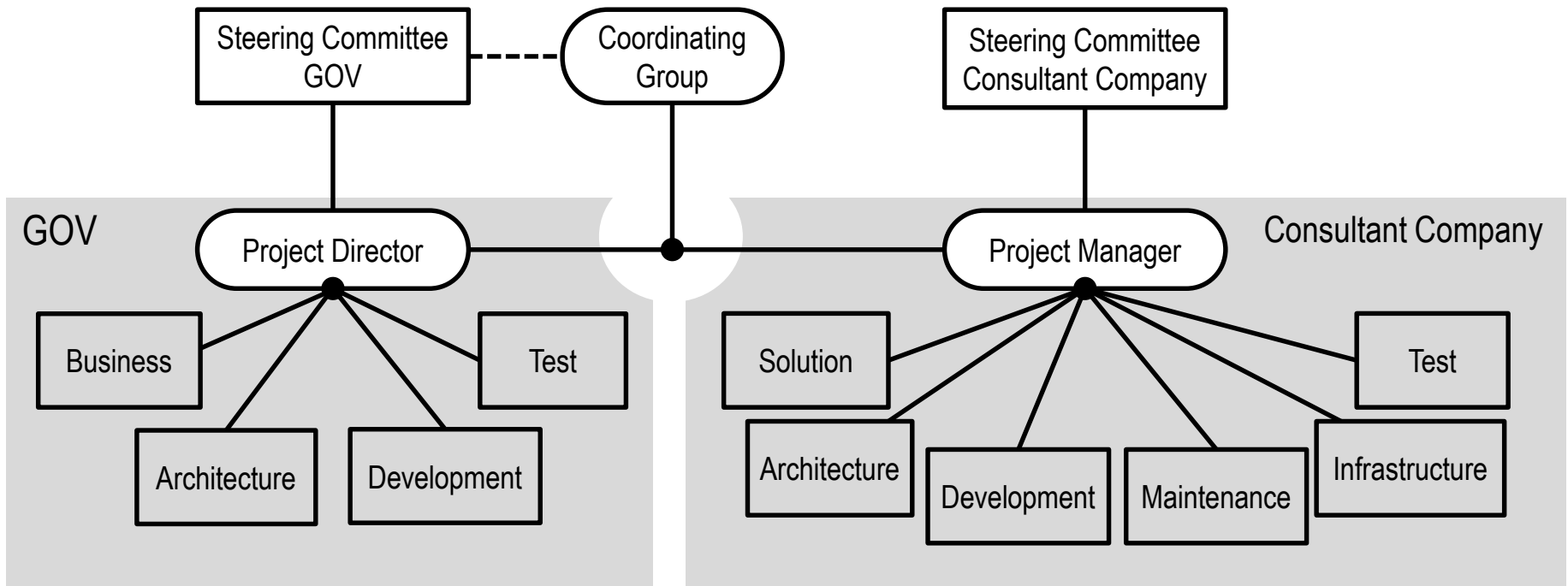
Mixed approaches: Water-Scrum-Fall



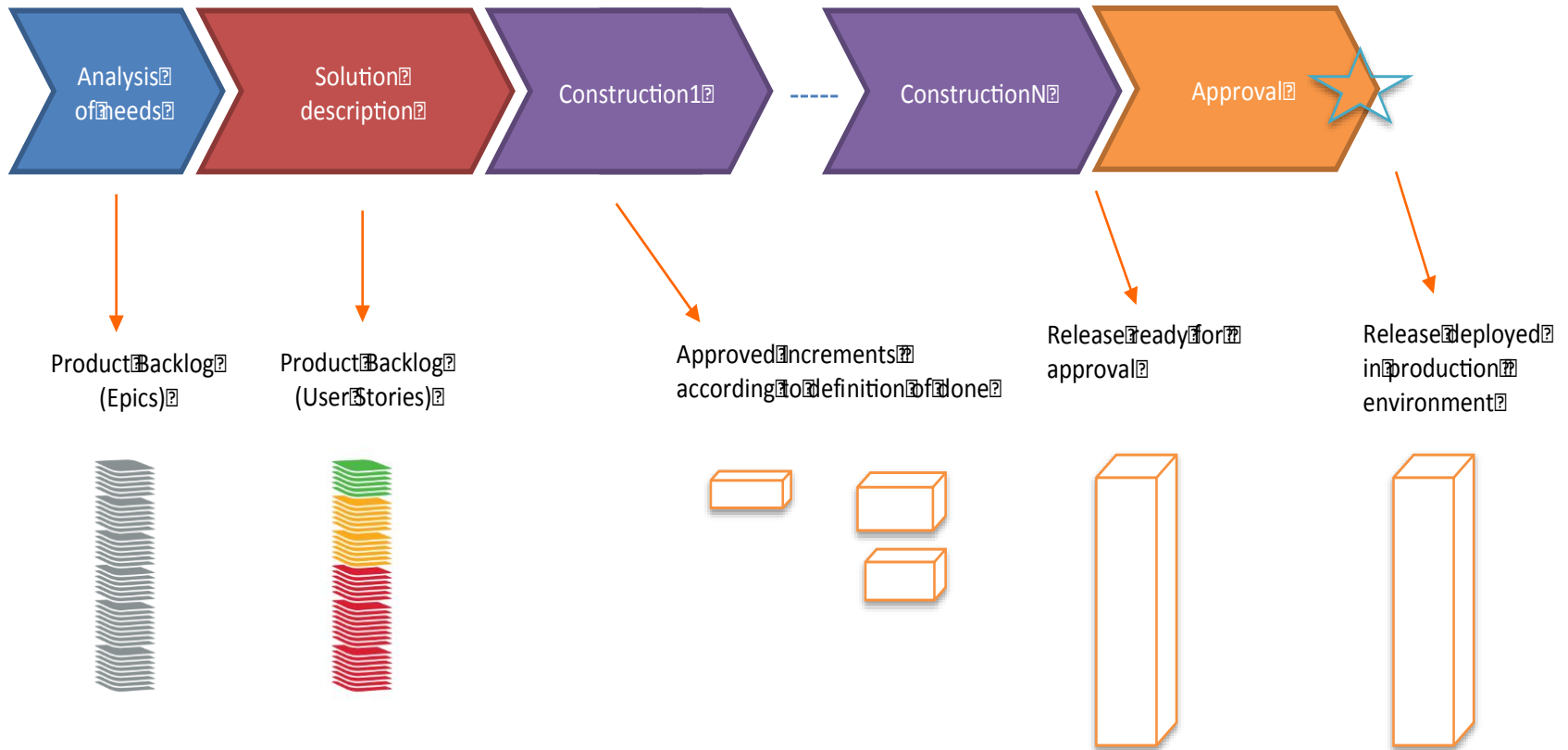
Schlauderer et al. (2015)

EXAMPLE FROM LARGE- SCALE IT-PROJECT IN PUBLIC SECTOR

Organization structure

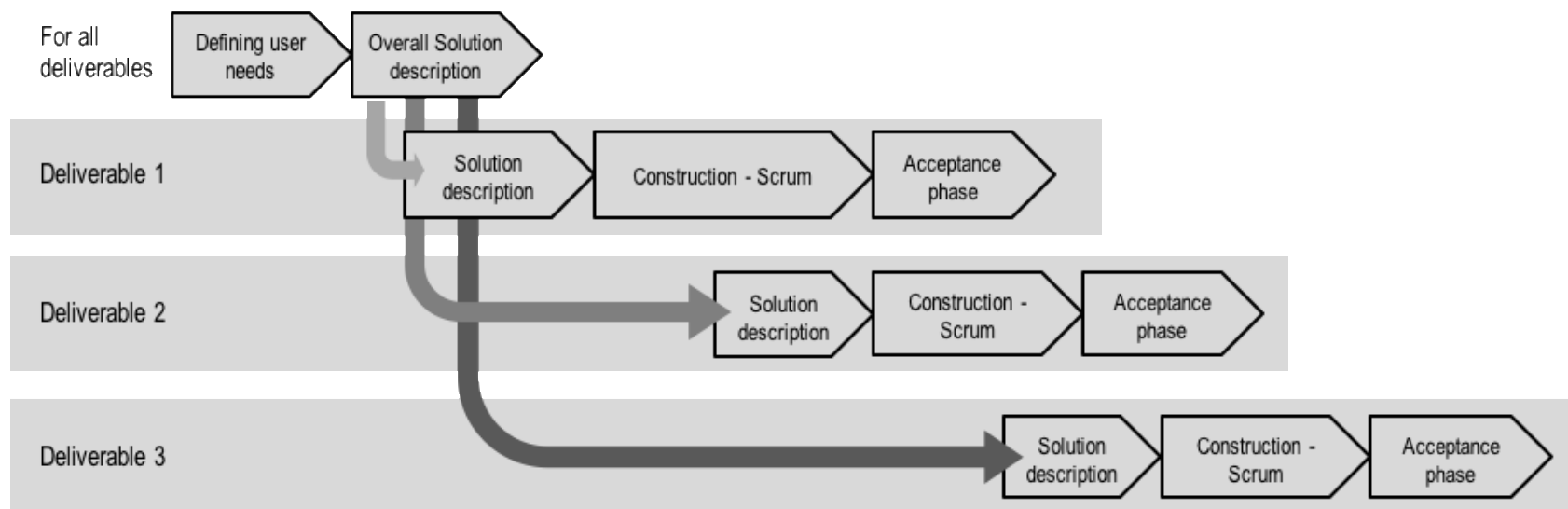


Phases in a large-scale IT-project



Source:: Kjetil Rød, Sopra Steria

Working on three deliverables in parallel



Project set up

- Large-scale public sector project (The ‘Beta project’, see Dingsøy et al., 2017)
- 5 large scrum teams
 - 2-3 Developers, UX designer, software architect, test responsible, functional responsible
- 3 deliverables over a period of nearly 4 years
- One dedicated integration team
- Additional meetings and roles: Architecture meetings, Bug-board, ready-to-sprint process, Tornado meetings, technical champions.

Complex dependencies

- Coordination mechanisms across teams just as important as coordination mechanisms within teams
- Project manager needed to focus on external stakeholders and coordinate with other projects

“When the other system being developed was down, it almost stopped the entire project. There was a tight coupling between this other system and our project, which we had not accounted for initially. We realized that this system needed to follow the same production schedule as our project – although they were not at all part of the official project.” —Software Architect, GOV

Organization structure is never perfect – needs continuous coordination across teams

- Continuous coordinating:

“By and large [coordination] is ad hoc. It was common practice to just walk over to each other [other teams] to discuss and solve issues there and then. And it was also a common understanding that such issues needed to be solved at once. And if [everyone] did so, this would certainly reduce the frictions between teams.” — Developer, ConsultGroup