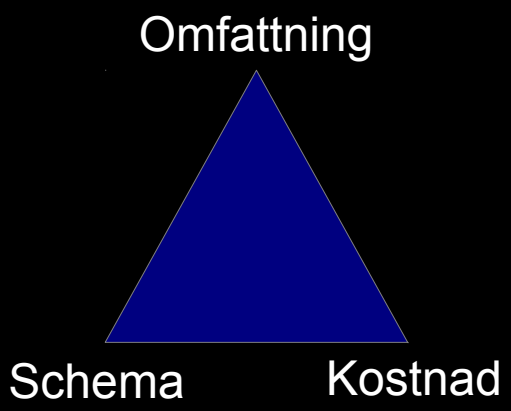
TNM094 – Medietekniskt kandidatprojekt

Projekthantering

Projekthantering

- Projekt
  - Samlat arbete med specifikt mål
  - Har begränsade resurser
  - Begränsat i tid (start- och slutdatum)
- Ledningens mål
  - Uppfylla målen
  - Levererat i tid och inom budget
- Projekthantering är kontinuerlig
  - Påbörjas med planeringen
  - Sker genom möten och avstämningar
  - Avslutas med ett avslutningsmöte



[Pfleeger]

Projektanalys – olika strukturer

- Work Breakdown Structure (WBS)
  - Identifiera och lista alla nödvändiga delar och steg
  - Hitta bästa ordning för dessa steg
- Organization Breakdown Structure (OBS)
  - Identifiera nödvändig kompetens och passande personer
  - Distribuera projektet mellan ledare och deras team
  - Dela ut ansvar och befogenheter
- Cost Breakdown Structure (CBS)
  - Identifiera och klassificera kostnader i projektet
  - Balansera kostnad mot värde och inkomst

o s v

[Pfleeger]

Work Breakdown Structure

- Hierarkisk nedbrytning av allt arbete
  - T ex faser, steg och aktiviteter
- Kan byggas top-down eller bottom-up
- Allt arbete summeras till 100 %

[Pfleeger]

Work Breakdown Structure

- 1 Krav
  - 1.1 Elicitation
    - 1.1.1 Intervjuer med intressenter
    - 1.1.2 Analysera föregående systems dokumentation
    - 1.1.3 Requirements modelling
  - 1.2 Analys
    - 1.2.1 Stabilitetsanalys av krav
    - 1.2.2 Mognadsanalys av krav
    - 1.2.3 Validering och verifiering av krav
- 2 Tekniktest och initial prototyp
  - 2.1 Teknikval
    - 2.1.1 Urval av tekniker att testa
    - ...

[Pfleeger]

Planering av arbetet

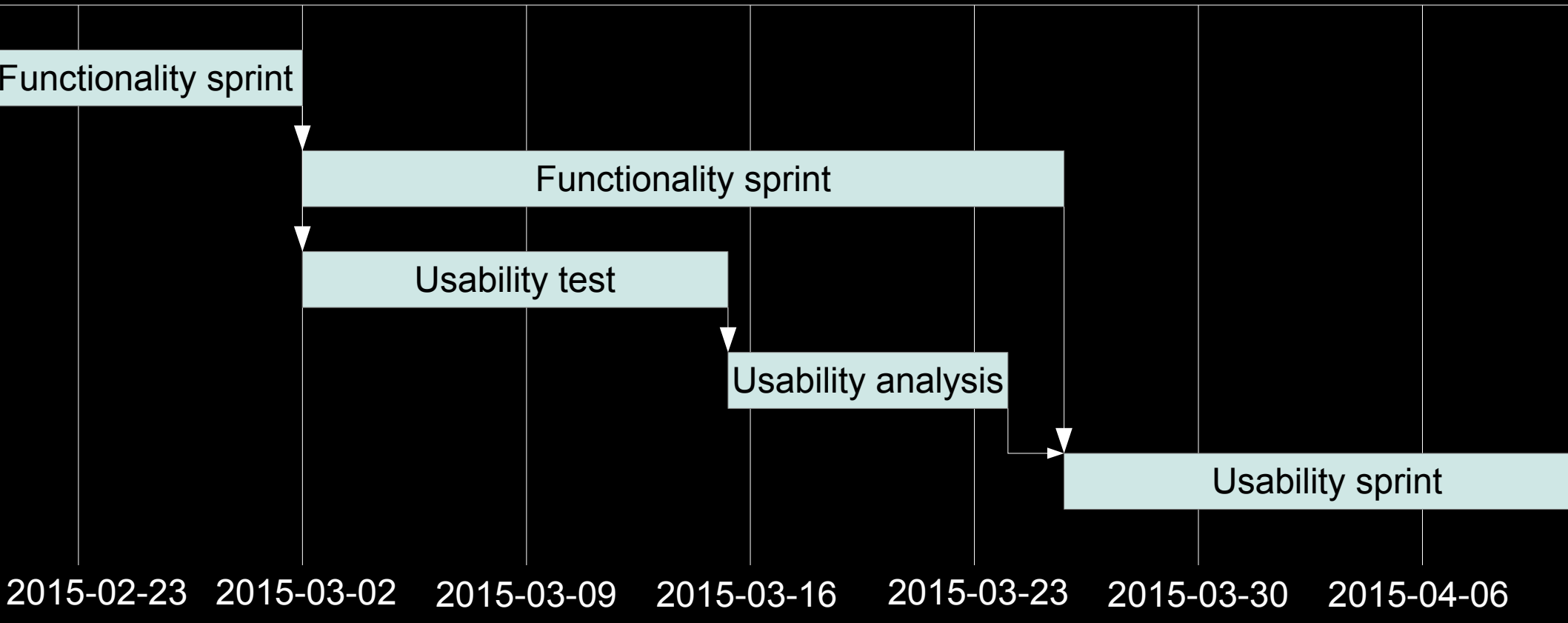
- Aktiviteter
  - Vad ska göras
- Leverabler
  - Vad ska produceras
  - Dokument, funktioner, demonstrationer, kvalitet, etc
- Milstolpar
  - Viktiga datum
  - Vissa beslut ska vara fattade
    (avbryt, ändra plan, justera budget, etc.)
  - Vissa aktiviteter ska vara genomförda
  - Vissa leverabler ska vara levererade

[Pfleeger]

Att planera Agil utveckling

- Icke-linjär utvecklingsprocess
  - Skapa en övergripande linjär plan
    - från start-datum till slut-datum
    - planera in milstolpar och leverabler
    - skapa boxar för sprintar
  - Per sprint
    - bedöm sprintens behov av resurser (typ och mängd)
    - planera in tillgängliga och nödvändiga resurser (reservationer)
- Skapa procedurer och rutiner för allt iterativt arbete
    - vad måste göras och i vilken ordning för varje bit arbete
    - skapa kö för arbetet som ska göras

Att planera Agil utveckling



Att planera Agil utveckling

- Scrum hanterar inte start och slut
  - i teorin slutar varje sprint med ett fungerande system!
    → continuous integration (CI)
  - men hur kommer vi igång?
  - men hur säkerställer vi att allt är klart i tid?
- En del förespråkar “Sprint 0”
  - skapa initiala krav och planera dem i alla sprintar
  - skapa tekniktest för alternativa tekniker
  - implementera övergripande struktur
  - kan man skriva projektplan före Sprint 0?

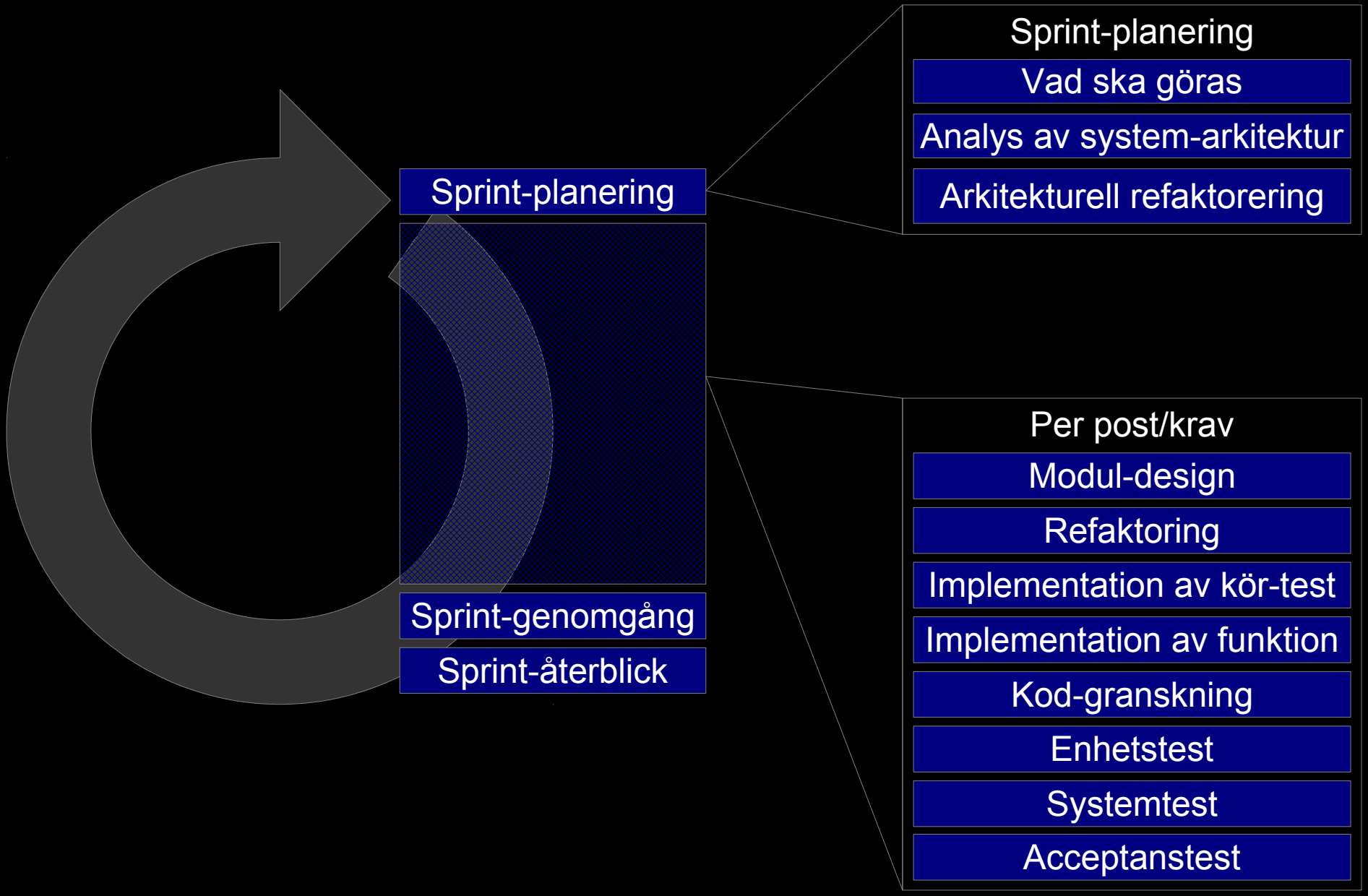
Att planera Agil utveckling

- Andra trick
  - Leverera flera små program istället för full CI
    - prototyp
    - konceptdemonstration
    - tekniktest

    eller till och med
    - prestandaanalyser
    - tekniska rapporter / forskningsrapporter / litteraturstudier

  - Det är varje post som specificerar vad som är dess leverabel

Exempel på plan för Agil utveckling



Viktiga aspekter vid planering

- Prioritera
  - Kostnad, schema, funktioner och kvalitet
  - Baserat på uppskattning av arbete
    och uppskattning av värde
  - I Agil utveckling: vilken post är mest kostnadseffektiv

- Håll planen inom budget
  - Givna ramar ska (kan?) inte brytas
    - Minska ambitionerna?
    - Ändra tillvägagångssätt?
  - Alternativt: omförhandla ramarna

[Pfleeger]

Skattning av arbete

- Expertbedömning
  - Informell (t ex Chief programmer)
  - Överenskommen (t ex planning poker)
  - Formaliserad (t ex viktade bedömningar)
- (Objektiva tekniker
  - Algoritmiska modeller – räkna utifrån tabeller
  - Machine learning – automatisk uträkning
  - Dynamiska modeller – simulera projektet)

[Pfleeger]

Fel i uppskattning av arbete

- Dålig kunskap om OH (overhead)
  - gruppdynamik, projektledning/-hantering, möten
- Dålig kunskap om det arbete som krävs
  - missbedömning av komplexitet, missade steg/uppgifter
  - okänd produktivitet

- Var förberedd
  - Uppskattningen kommer *alltid* vara mer eller mindre fel
  - Ha en buffert (slack) för signifikant underskattning (80% beläggning är vanligt)

[Pfleeger]

Organisation

- Viktiga aspekter
  - Hur stort bör vårt team vara?
  - Ska man ha del-team?
  - Vilken ledarstruktur bör man ha?
  - Vem är ansvarig för vad?
  - Hur ofta träffas vi och vad ska diskuteras?

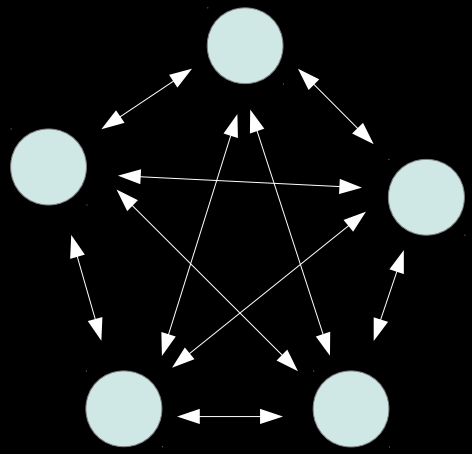
[Pfleeger]

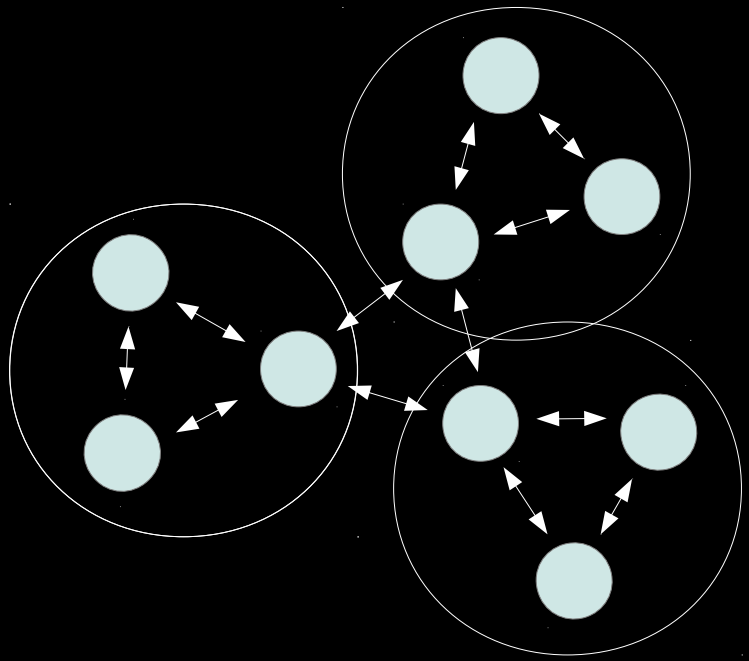
Ledarstruktur

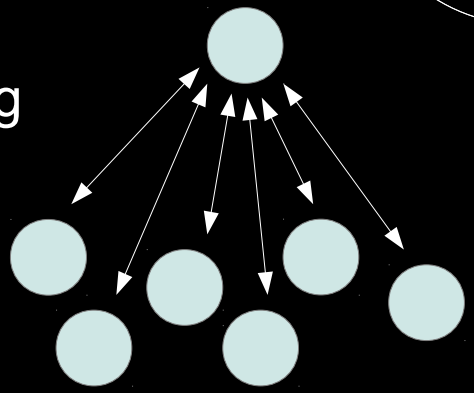
- Team och andra grupper
  - kommunikationsvägar

$$L = N(N-1)/2$$

- Strukturer
  - Hierarkisk
    - t ex Chief programmer team
  - Semi-hierarkisk
    - t ex Scrums of scrum
  - Platt
    - t ex Egoless programming







[Pfleeger]

Ledarstruktur

- Viktiga faktorer
  - Hur säkert teamet är på projektet
  - Hur mycket ny teknik som är inblandad
  - Storleken på projektet
  - *Viss* avvägning mellan struktur och kreativitet

[Pfleeger]

Ansvar

- Ansvar
  - Befogenhet går hand i hand med ansvar
  - Bara en person är ansvarig vid en viss tidpunkt
  - Ansvar som inte är formellt är inte ett ansvar
- Redundans
  - Reservansvarig person vid behov
- I ert projekt
  - Se till att någon har formellt ansvar för varje viktig aspekt
  - Denne sätter upp rutiner och kontrollerar att alla gör sitt

Avstämningsmöten

- Kontinuerlig framstegs- och funktionskontroll
  - Vad avviker från planerna?
  - Vad behöver vi göra för att fortsätta att följa planen?
  - Uppdatera plan och schema utifrån verkligheten!

- Typisk agenda
  - Vad har hänt sedan senaste mötet
  - Vad kommer att hända till nästa möte
    - Vem är inblandad och hur mycket
    - Arbetsbelastning, risker, distrationer, etc

[Pfleeger]

Andra viktiga aspekter

- Risk-hantering
  - hantera oplanerade händelser
    (sjukdom, försenade leveranser, stöld, etc)
- QA-hantering
  - säkerställa den övergripande kvalitén
    (säkerhet, felfrekvens, möta krav, etc)

[Pfleeger]

Risk-hantering

- En risk är en framtida händelse
  - som får en ***negativ effekt***
  - som ***kan inträffa*** med viss sannolikhet
  - som ***kan påverkas***

- Risk-exponering
  - Den statistiska kostnaden för att risken existerar

$$E(e) = P(e)C(e)$$

- Risk Management Plan
  - identifierar risker
  - listar sätt att undvika eller hantera problemen

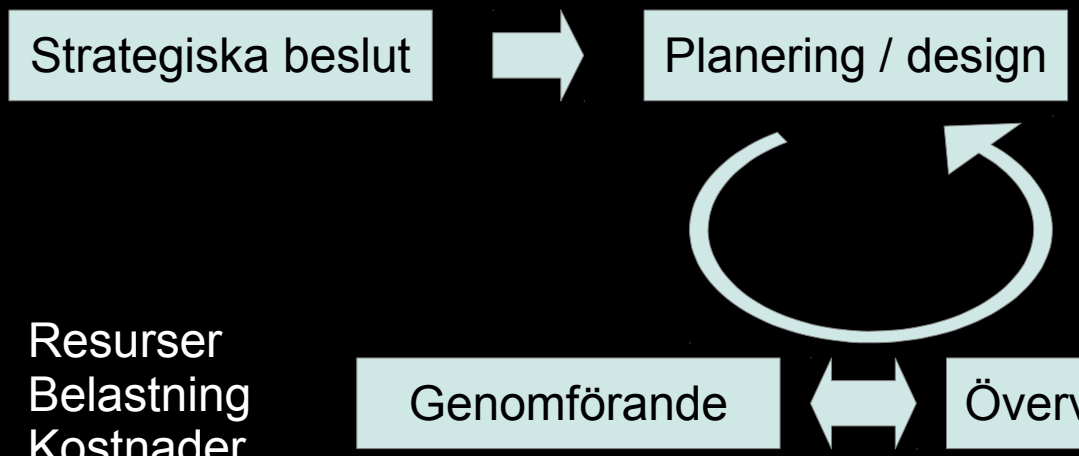
[Pfleeger]

QA – Quality Assurance

- Säkerställa kvalitet och stabilitet
  - procedurer, rutiner och verktyg
  - test-planering och -hantering
- Systemsäkerhet
  - analys av systemkarakteristik, hot och sårbarhet
  - övergripande mål och plan
  - procedurer, rutiner och verktyg för säkerhetsarbetet

[Pfleeger]

Projekthantering



Resurser
Belastning
Kostnader
Kvalitet
Funktioner
...

Möten
Rutiner
Test
Rapporter
Mätning
...

Projektplan

- Överenskommelse
  - kommunicerar planen med kund och ledningen
  - visar att vi har vad som behövs
  - överenskommelse mellan utvecklarna (visar hur vi ska arbeta)
- Projektstruktur
  - utgör grunden för utvecklarna att arbeta utifrån
  - visar vem som förväntas göra vad och när

[Pfleeger]

Projektplan

- Beskrivning av projektet, syfte och mål
- Omfattning och begränsningar
- Tidsplan, schema, milstenar
- Organisation, resurser, etc
- Teknisk beskrivning
- Infrastruktur och rutiner för systemutveckling
- Planer för QA, leverans, extern kommunikation, etc

[Pfleeger]

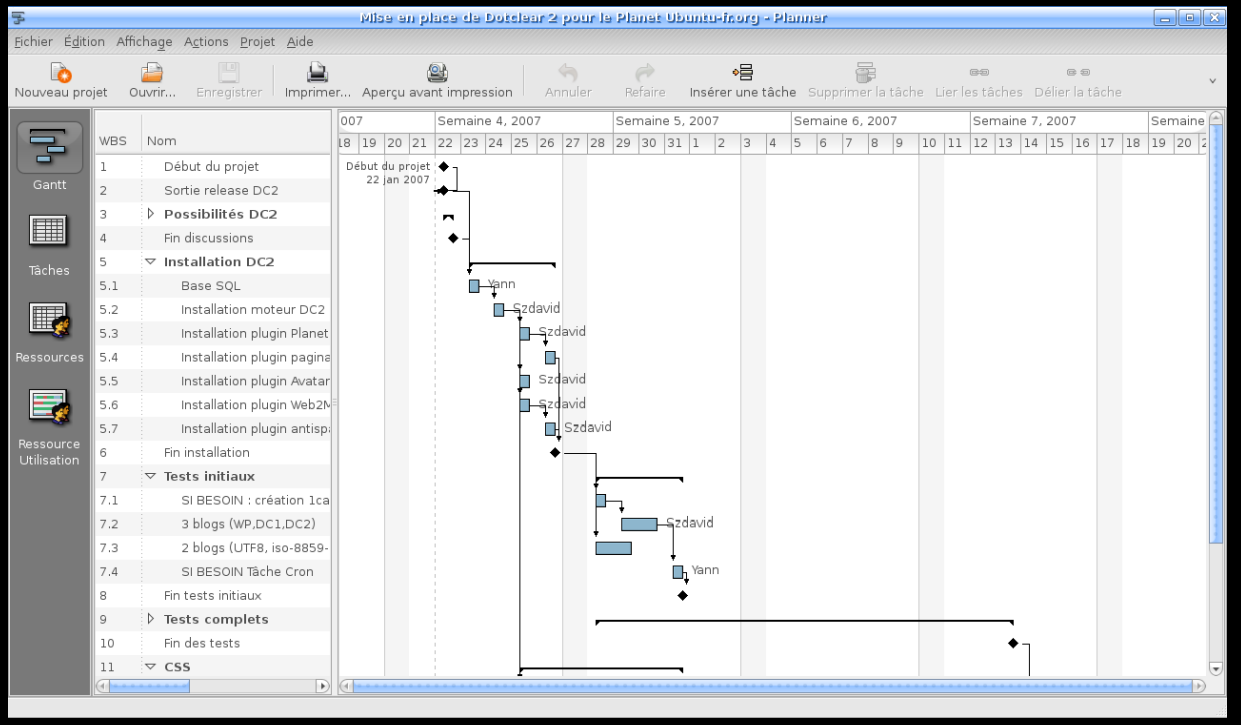
Verktyg och infrastruktur

- Projekthantering
  - Schemaläggning och resursläggning
  - Dokumenthantering och -spårning
- Utvecklingsvertyg
  - Versions-hantering (Subversion, GIT, etc.)
  - Systemstöd (IDE, profilering, statisk kodanalysator, etc.)
  - Automatisk dokumentationsgenerator (Doxygen, etc.)
  - CI-server
- Ramverk och APIer

[Pfleeger]

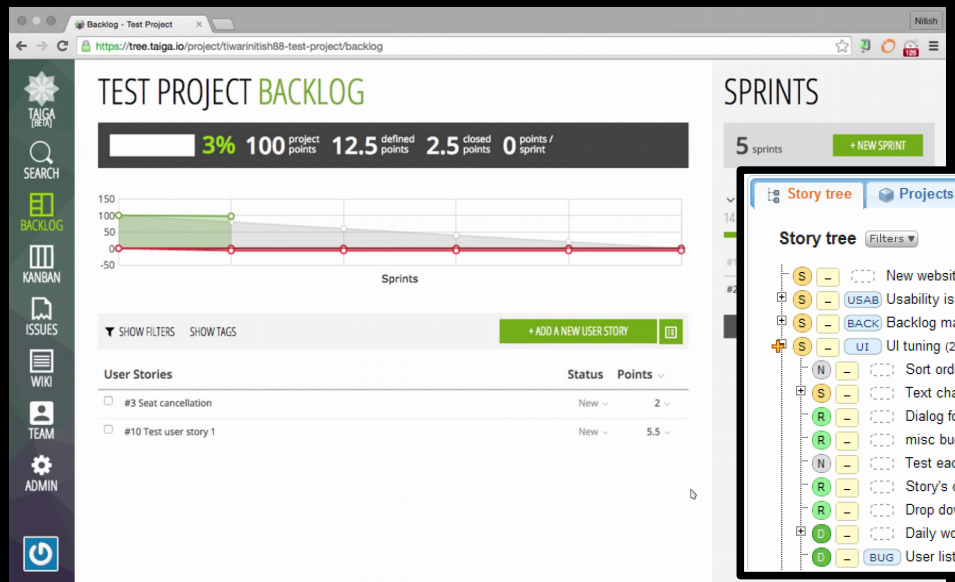
Projekthanteringsverktyg

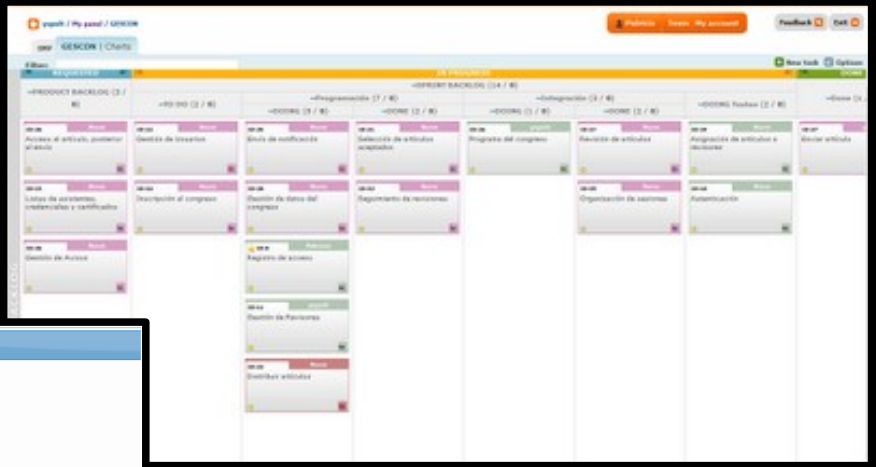
- Linjära verktyg – från start till slut
  - MS Project, LibreOffice Project, Calligra Plan
  - Spåra uppgifter, resurser, kostnader och kontering
  - Schemaläggning via kritisk-sekvens-analys (CPA/CPM)

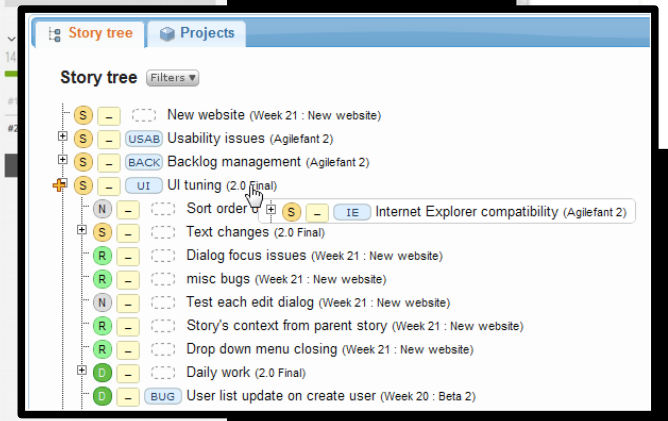


Projekthanteringsverktyg

- Verktyg för löpande utvecklingsarbete
  - Taiga.io, Trello, GitHub, Pivotal Tracker, SeeNowDo, Bitbucket, Agilefant
  - Kravhantering / backlog / problemlista
  - Resursallokering, arbetsövervakning







Dokument-hantering

- Inte så lätt som det låter
  - Stora projekt producerar en stor mängd dokument
  - Bra verktyg hjälper till med hanteringen av dessa
  - Bra plan underlättar enhetligt arbete med dokument
- Verktyget bör stödja
  - Versionshantering
  - Distribution, publicering, sökning, etc
  - Kollaborativt arbete – samtida editering
  - Gruppering och korsreferenser

[Pfleeger]

Några termer om dokument

dokument: en skriftlig källa eller urkund som innehåller information.

statiskt dokument: ett dokument som beskriver någonting knutet till en specifik tidpunkt, såsom mötesprotokoll och statusrapporter.

levande dokument: ett dokument som ständigt uppdateras för att hållas aktuellt. Det är komplett och aktuellt, inte under uppbyggnad.

evergreens: dokument som är så generellt skrivna att de gäller under en längre tid och inte behöver uppdateras.

spårbarhet: möjligheten att följa en aspekt mellan flera olika dokument, såsom en funktion från krav till dess verifierande test, via design och implementation.

[Pfleeger]

Utvecklingsverktyg

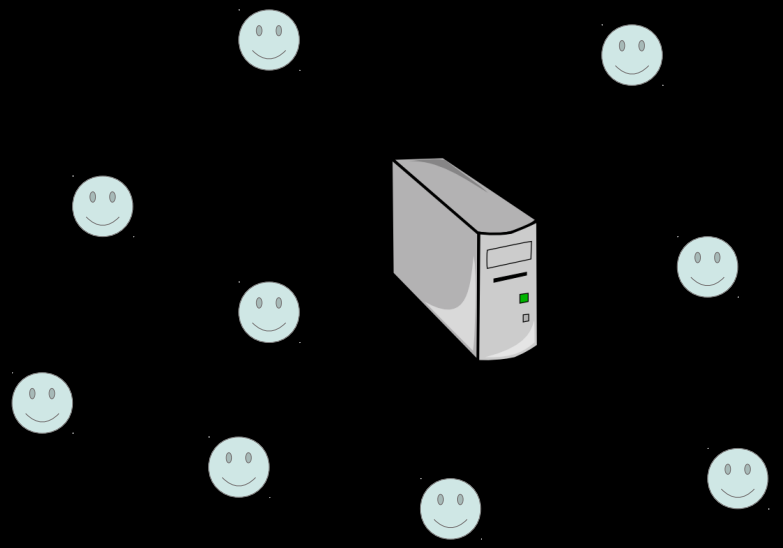
- IDE – integrerad utvecklingsmiljö
  - kodkomplettering, implicit referens-manual
  - automatisk indentering, kodformatering
  - debuggning, profilering, realtids-analys
- Profilering och analys
  - realtids prestanda-analys
    (cache-användning, CPU-användning, flaskhalsar, etc)
  - statisk kod-analys
    (läsbarhet, problematisk kod, ambivalens, etc)

[Pfleeger]

Ramverk och API:er

- Egenutvecklad kod
  - Kostnad för utveckling + framtida underhåll
  - Ibland mer *applikationsorienterad* kod
- Tredjeparts-API:er och -ramverk
  - Strukturerad, testad och dokumenterad
  - Oftast mer *funktionsorienterad* kod
  - Du betalar för kvalité och underhåll
- Ramverk
  - SAP, Qt, Java EE, etc
  - Använder domän-specifika design-mönster

CI-server (Build server)



- Hjärtmonitor för projektet
  - Automatisk kompilering
  - Automatisk analys
  - Automatiska test
  - Automatisk paketering
  - Även för flera konfigurationer och plattformar
  - Rapporterar fullständig återkoppling
- Viktigt att tänka på
  - Bygg en organisation och rutiner kring CI-servern
  - Vad gör vi när CI-server rapporterar fel?