

Kontrakter og prosjektstyring i store, smidige IT-prosjekter

Jo Hannay
Simula Research Laboratory

Prosjekt PLASMA
**Planlegging, samhandling og styring i store,
smidige IT-prosjekter**

PROMIS AS
Simula Research Laboratory
Statens pensjonskasse,
Accenture,
Steria,



Temaer vi ser på:

- 1. Kontraktstandarder i store smidige prosjekter**
- 2. Leveranseplanlegging i store smidige prosjekter**
- 3. Produktivitetshemmere i store smidige prosjekter**



Forskning:

Gode ideer

Forskning

Modeller
Verktøy
Metoder

Evaluering:

Hva skjer i praksis?

Bør verktøy og metoder støtte eller erstatte eksisterende praksis?

(Videre)utvikling av verktøy og metoder

Arbeidslivstudier
"Smidig" forskning
Akademisk/Praksis
Ekspertene følger ikke "reglene"



Temaer vi ser på:

- 1. Kontraktstandarder i store smidige prosjekter**
- 2. Leveranseplanlegging i store smidige prosjekter**
- 3. Produktivitetshemmere i store smidige prosjekter**



1. Kontraktstandarder og store smidige prosjekter



Kontraktstandarder i store smidige prosjekter



Kontraktstandarder :

DIFI: Statens standardavtale (programutviklingsavtalen) - kundeinteresser

IKT-Norge: Standardavtale for systemutvikling - leverandørinteresser

Dataforeningen: PS2000 – kunde/leverandør-interesser, forskningsbasert

Forskningsprogrammet PS2000:

Europas største forskningsprogram innen prosjektledelse, under ledelse av NTNU og SINTEF

- **Erfaringsinnsamling fra større IT-prosjekter (NTNU)**
- **Utarbeidelse av ny kontraktsstandard med veiledning (PROMIS med Selmer)**
- **Tildelt forskningsmidler gjennom SkatteFUNN (Norges Forskningsråd) for videreutvikling**
- **Dataforeningen har overtatt forvaltningsansvaret for PS2000 gjennom Faggruppe for IT-kontrakter som ledes av PROMIS**



Kontraktstandarder i store smidige prosjekter



Kontraktstandarder :

DIFI: Statens standardavtale (programutviklingsavtalen) - kundeinteresser

IKT-Norge: Standardavtale for systemutvikling - leverandørinteresser

Dataforeningen: PS2000 – kunde/leverandør-interesser, forskningsbasert

PS2000 er designet til å...:

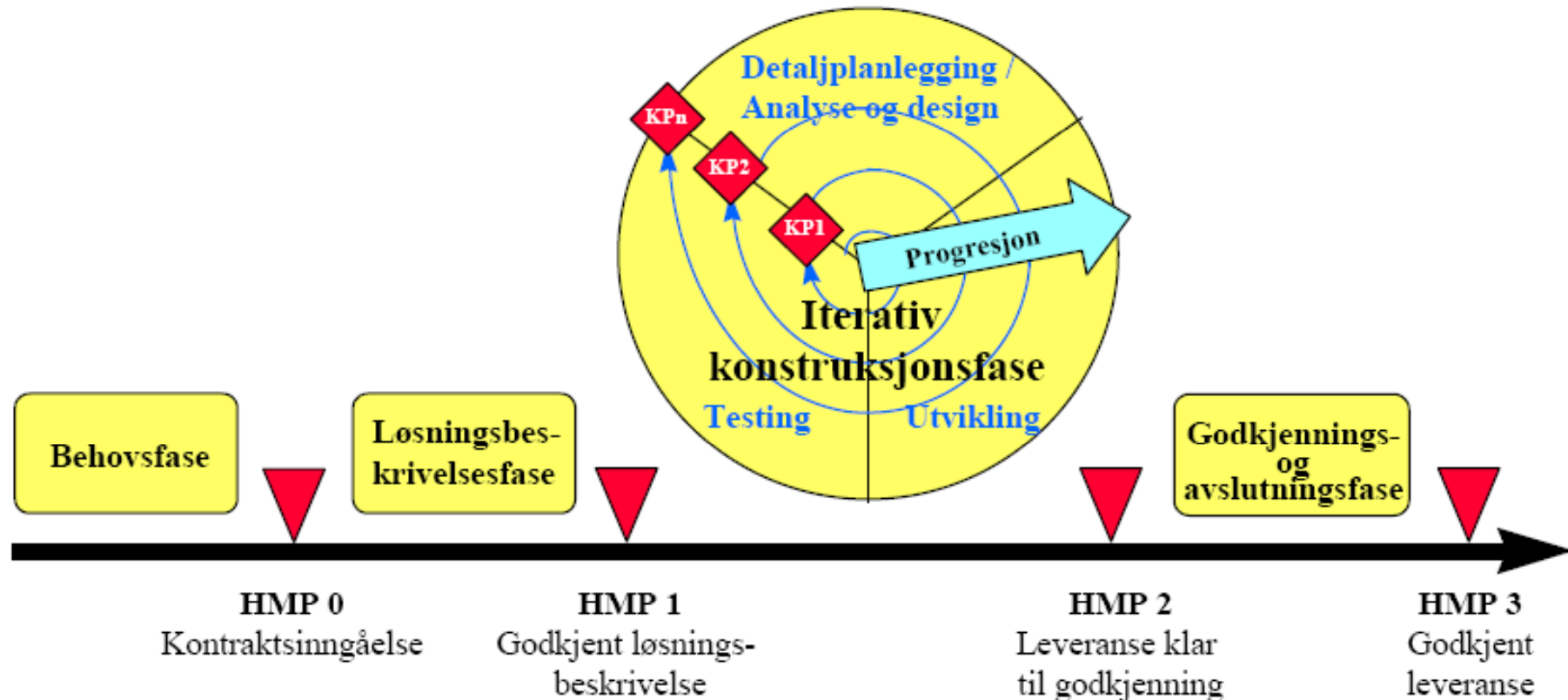
- ...øke mulighetene for at begge parters interesser er ivaretatt og balansert. (PS2000 er utviklet av kunder og leverandører i samarbeid.)
- ...tilrettelegge for å fange opp læring under prosjektgjennomføring. (Gjennomføringsmodellen skal legge til rette for integrert samarbeid mellom partene.)
- ...tilrettelegge for bruk av motiverende økonomiske incentivordninger, slik at risiko blir delt. (En usikkerhetsanalyse skal legges til grunn.)
- ...tilrettelegge for iterativ gjennomføringsmodell, versjon 3.0 for Scrum.



Kontraktstandarder i store smidige prosjekter



Kontraktstandarder :



- ...tilrettelegge for iterativ gjennomføringsmodell, versjon 3.0 for Scrum.



Kontraktstandarder i store smidige prosjekter



Kontraktstandarder :

DIFI: Statens standardavtale (programutviklingsavtalen) - kundeinteresser

IKT-Norge: Standardavtale for systemutvikling - leverandørinteresser

Dataforeningen: PS2000 – kunde/leverandør-interesser, forskningsbasert

PS2000 er designet til å...:

- ...øke mulighetene for at begge parters interesser er ivaretatt og balansert. (PS2000 er utviklet av kunder og leverandører i samarbeid.)

- ...tilrettelegge for å fange opp læring under prosjektgjennomføring. (Gjennomføringsmodellen skal legge til rette for innbyrdes samarbeid mellom partene.)

- ...tilrettelegge for bruk av motiverende økonomiske incentivordninger, slik at risiko blir dekket. (En usikkerhetsanalyse skal legges til grunn.)

- ...tilrettelegge for iterativ gjennomføringsmodell, versjon 3.0 for Scrum.

Funker dette etter hensikten?



Kontraktstandarder i store smidige prosjekter



Forskningsspørsmål:

Hva er erfaringer med ulike kontraktstandarder (DIFI, IKT-Norge, PS2000) i store iterative utviklingsprosjekter Masterprosjekt, Farah Kahn

Fokus på PS2000:

- 1. Hva er bakgrunnen for at bedrifter velger å ta i bruk PS2000?**
- 2. Hva er erfaringene med PS2000 i store iterative prosjekter?**
- 3. Hva er fordelene og ulempene med risikodeling i PS2000?**
- 4. Bidrar PS2000 til å løse utfordringer i andre kontraktstandarder?**



Kontraktstandarder i store smidige prosjekter

Intervjuet 9 PS2000-brukere (tilfeldig valgt fra Dataforeningens lister)

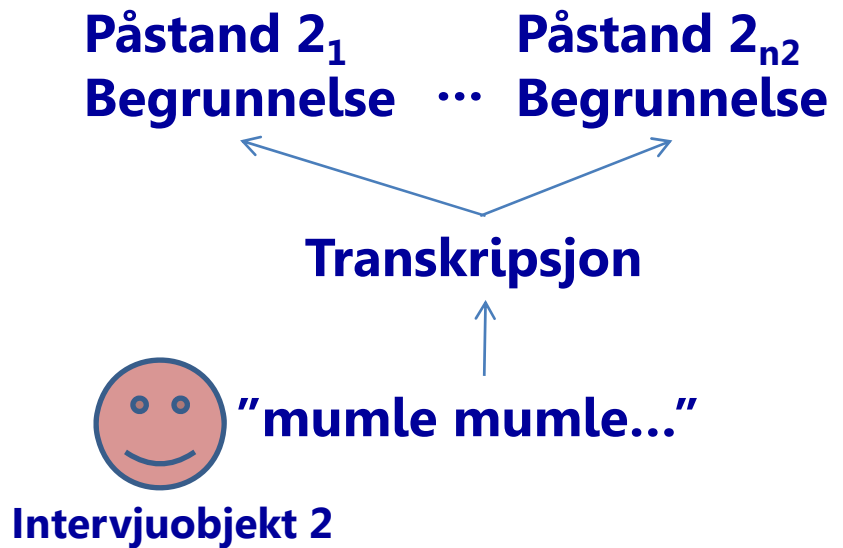
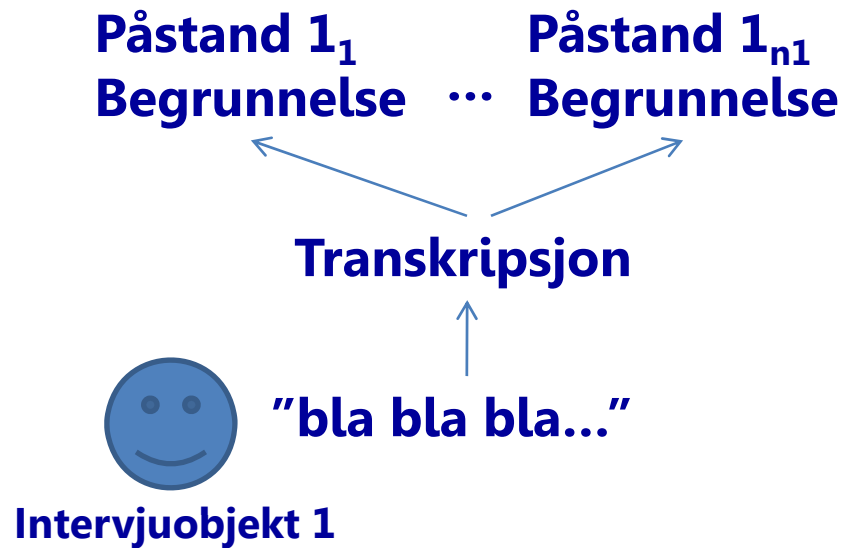
	Oppgave	Aktør
1	Prosjektleder	Kunde
2	Rydde opp i avtalestrukturen	Kunde
3	Ansvarlig for anskaffelsesprosessen og kontraktsoppfølging	Kunde
4	Oppfølging av leverandør og kontrakter	Kunde
5	Utvikler	Kunde
6	Oppfølging av leverandør	Kunde
7	Prosjektleder	Leverandør
8	Prosjektleder	Leverandør
9	Ansvarlig for oppfølging av kontrakter	Leverandør



Kontraktstandarder i store smidige prosjekter

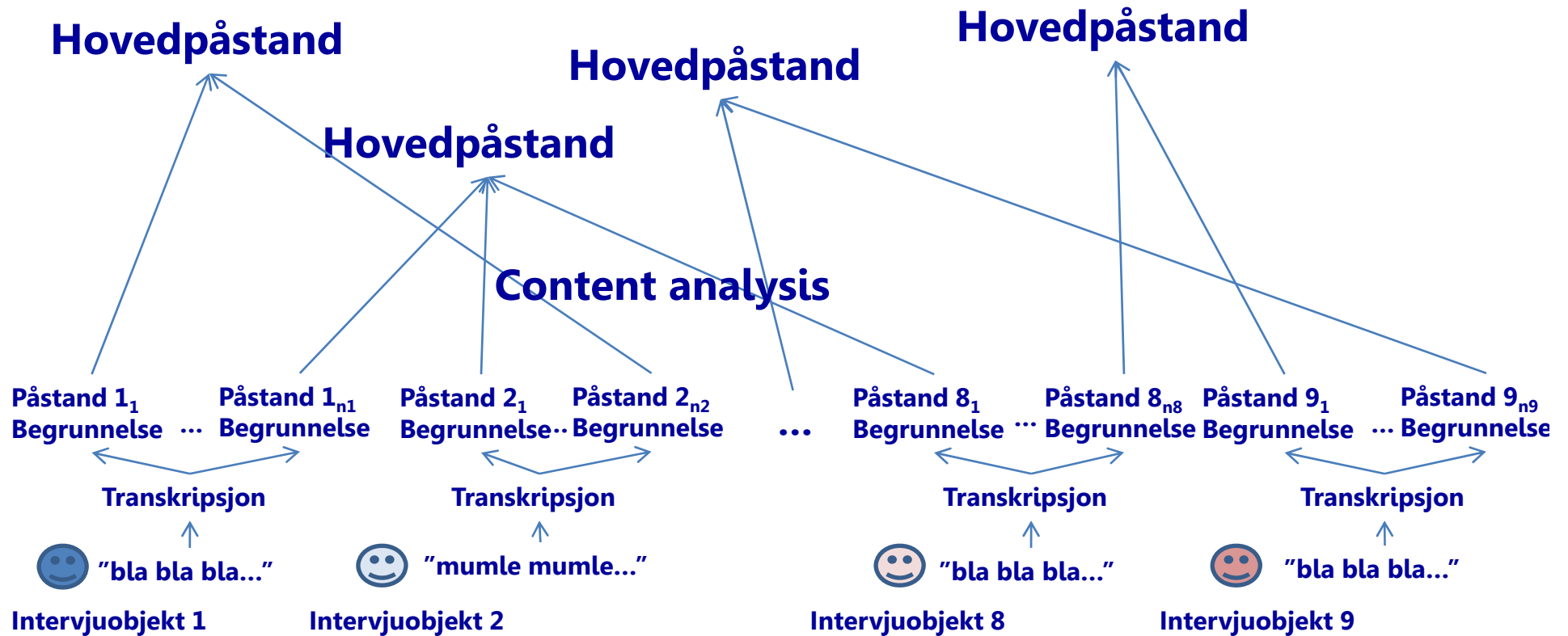
Intervjuene

**Litt som evidensbasert praksis,
men ikke helt**
(Dybå, Kitchenham, Jørgensen IEEE Software 2005)



Kontraktstandarder i store smidige prosjekter

Intervjuene



Kontraktstandarder i store smidige prosjekter



Hovedpåstander for PS2000-erfaringer



Kontraktstandarder i store smidige prosjekter

Årsaker til valg av kontraktstandard:

- PS2000 blir valgt fordi den støtter iterativ utvikling (5,2)
- PS2000 blir valgt fordi det er en balansert kontrakt (1,2)

Gjennomføring:

- PS2000 er gunstig for samarbeid, kontroll og åpenhet (3,1)
- Løsningsbeskrivelsesfasen i PS2000 er krevende (-,3)
- PS2000 øker bevisstheten rundt de tre elementene i prosjekttrekanten (1,1)
- Hovedstyrken til PS2000 er prosjektmetodikken (2,3)
- Det er få som er iterative i konstruksjonsfasen ved bruk av PS2000 (1,1)

Risikodeling:

- Risikodeling bidrar til tettere samarbeid og mer effektivitet (2,2)
- Incentivordningene er blant det mest positive i PS2000 (3,1)
- Få prosjekter nyter godt av incentiver (2,-)
- Administrasjon av målpris er krevende (1,2)

Annet:

- PS2000 løser utfordringer (1,1)
- Bruk av PS2000 krever en moden kunde (1,3)
- Enkelte ting er ikke tydelig regulert i PS2000 (2,-)
- PS2000 er komplisert (2,1)
- SSA er en ubalansert kontrakt (2,1)
- PS2000 kan også virke negativt for samarbeidet mellom partene i et prosjekt (1,3)



Kontraktstandarder i store smidige prosjekter



Svar på forskningsspørsmål:

Hva er erfaringer med ulike kontraktstandarder (DIFI, IKT-Norge, PS2000) i store iterative utviklingsprosjekter

Fokus på PS2000:

- 1. Hva er bakgrunnen for at bedrifter velger å ta i bruk PS2000?**
- 2. Hva er erfaringene med PS2000 i store iterative prosjekter?**
- 3. Hva er fordelene og ulempene med risikodeling i PS2000?**
- 4. Bidrar PS2000 til å løse utfordringer i andre kontraktstandarder?**



Kontraktstandarder i store smidige prosjekter



Svar på forskningsspørsmål:

Hva er erfaringer med ulike kontraktstandarder (DIFI, IKT-Norge, PS2000) i store iterative utviklingsprosjekter

Leverandører: "SSA er en ubalansert kontrakt som er altfor kundevennlig."

(Støtter Poppendieck x 2 "Lean Software Development: An Agile Toolkit", 2003)

"Leverandør har økonomisk risiko i SSA, men kunden risikerer å få produktet levert sent og med dårligere kvalitet ." (Støtter Poppendieck igjen)

"SSA legger mer til rette for fossefallsmetoden, og sier kun noe om hva som skal leveres, når det skal leveres og hvor mye det skal koste. Det sier ingen ting om prosessene og samarbeidet i et prosjekt, noe som har vært svært problematisk i ulike prosjekter de har hatt med SSA." (Støtter Sliger og Broderick, The Software Project Manager's Bridge to Agility. 2008)

Samtlige intervjuobjekter mener at SSA ikke er en passende kontrakt hvis prosjektet skal være iterativt.



Kontraktstandarder i store smidige prosjekter



Svar på forskningsspørsmål:

1. Hva er bakgrunnen for at bedrifter velger å ta i bruk PS2000?

”PS2000 er mer balansert kontrakt sammenlignet med SSA. Risikodeling og felles mål.”

Prosjektmetodikken. Syv av intervjuobjektene har valgt å bruke PS2000 fordi den understøtter iterativ metode. Noen av intervjuobjektene mener utviklingsmetodikken er hovedårsaken til kontraktbyttet.

Intervjuobjektene fra leverandørsiden er også alle enstemmige om at de vil ta i bruk PS2000 for neste prosjekt hvis de får velge. Kundene har derimot delte meninger



Kontraktstandarder i store smidige prosjekter



Svar på forskningsspørsmål:

2. Erfaringer med bruk av PS2000 i store prosjekter med iterativ modell?

Fem av intervjuobjektene: "hovedstyrken at det er tilrettelagt for iterativ modell."

"Men tilretteleggingen betyr ikke nødv. at prosjektet blir gjennomført iterativt."

Noen: "lang vei å gå før agile metodikk kan tilpasses utvikling i offentlig sektor på grunn av de faste rammene som er bestemt fra politisk hold."

Agile i offentlig sektor? Kontroll + smidig?? Hovedproblemstillingen!

Store prosjekter har flere nivåer. Prosjektet kan kjøres smidig på mer tekniske nivåer (kravene til informasjonsflyten og skjermbilder kan diskuteres iterativt). Det har fungert godt i organisasjonen hvor intervjuobjekt 1 jobber.

Det kan være spesielle organisatoriske og individuelle utfordringer som fører til dårligere resultater i offentlige prosjekter (Moløkken-Østvold, et al.: Management of Public Software Projects: Avoiding Overruns, 2005)



Kontraktstandarder i store smidige prosjekter



Svar på forskningsspørsmål:

3. Hva er fordelene og ulempene med risikodeling i PS2000?

Samtlige intervjuobjekter nevner risikodeling som noe av det mest positive i PS2000.

5 av intervjuobjektene: "samme båt" tettere samarbeid mellom kunde og leverandør.

(Fehr, E., A. Klein, and K. Schmidt, Kontrakts, Fairness, and Incentives. 2004, CESifo Group Munich; Eckfeldt, B., et al., Selling Agile: Target-Cost Kontrakt, 2005; Poppendieck, T. and M. Poppendieck, Implementing Lean Software Development: From Concept To Cash. 2007)

To er uenige: "Partene blir opptatt av sitt for å kunne levere tidsnok, og mindre villige til å hjelpe hverandre. Mister fokuset på helheten, noe som skader samarbeidet mellom kunde og leverandør i det lange løp. "

"Målpris krever mye administrasjon i tillegg til at det er en komplisert jobb å beregne insentiver. De etterlyser en enklere veiledning på hvordan dette kan gjøres."

Leverandør: "kunder ser på PS2000 som en fleksibel avtale hvor man kan tillate seg flere endringer underveis uten å registrere dette som en endring. Mulig manglende detaljer rundt endringshåndtering."



Kontraktstandarder i store smidige prosjekter



Svar på forskningsspørsmål:

4. Bidrar PS2000 til å løse utfordringer i andre kontraktstandarder?

PS2000 er designet til å...:

- ...øke mulighetene for at begge parter interesser er ivaretatt og balansert. **Ja, men ikke alltid (f eks ved heving av kontrakt). Mye er uspesifisert.**
- ...tilrettelegge for å fange opp læring under prosjektgjennomføring. (Gjennomføringsmodellen skal legge til rette for integrert samarbeid mellom partene.) **Ja, men kan også føre til fokus på egne mål. Kunden må være kunnskapsrik.**
- ...tilrettelegge for bruk av motiverende økonomiske modeller i form av incentivordninger, slik at tids- og kostnadsbesparelser kommer begge parter til gode. (En usikkerhetsanalyse skal legges til grunn ved valg av spesifikke incentiver.) **Ja, men det er sanksjonene som teller, og incentivmodellen er vanskelig. (Usikkerhetsanalysen blir ikke brukt.)**
- ...tilrettelegge for iterativ gjennomføringsmodell, versjon 3.0 for Scrum. **Ja, men prosjektet blir ikke nødvendigvis smidig.**



Kontraktstandarder i store smidige prosjekter

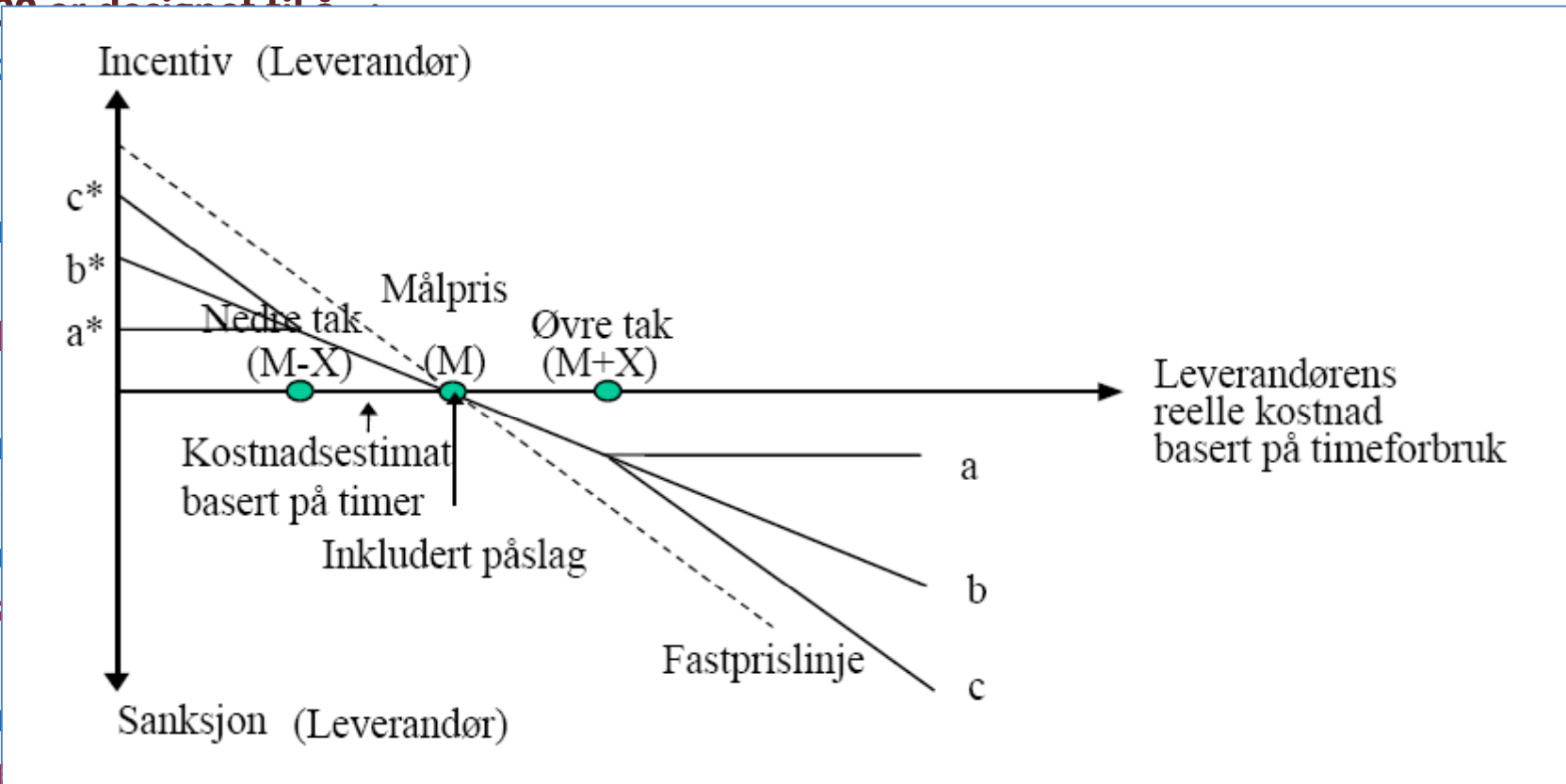


Svar på forskningsspørsmål:

4. Bidrar PS2000 til å løse utfordringer i andre kontraktstandarder?

PS2000 er designet til å

- ...øke alltid
- ...til (Gjenmen l
- ...til incen usikk sanks
- ...til prosje



Temaer vi ser på:

- 1. Kontraktstandarder i store smidige prosjekter**
- 2. Leveranseplanlegging i store smidige prosjekter**
- 3. Produktivitetshemmere i store smidige prosjekter**



Temaer vi ser på:

2. Leveranseplanlegging i store smidige prosjekter

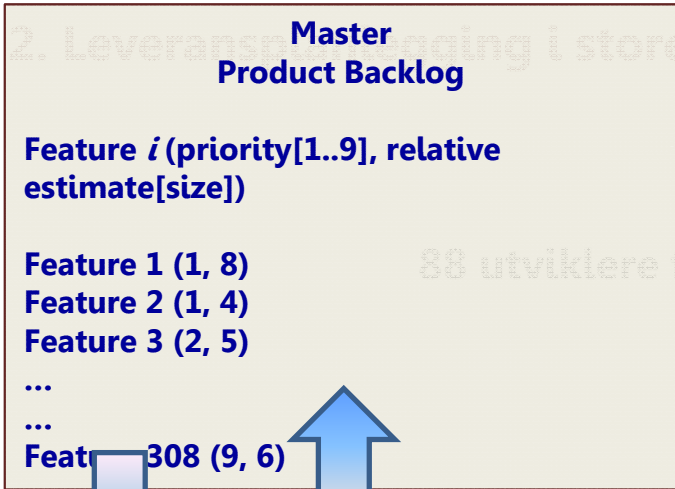


2. Leveransplanlegging i store smidige prosjekter



Sliger and Broderick: The Software Project Manager's Bridge to Agility, 2008

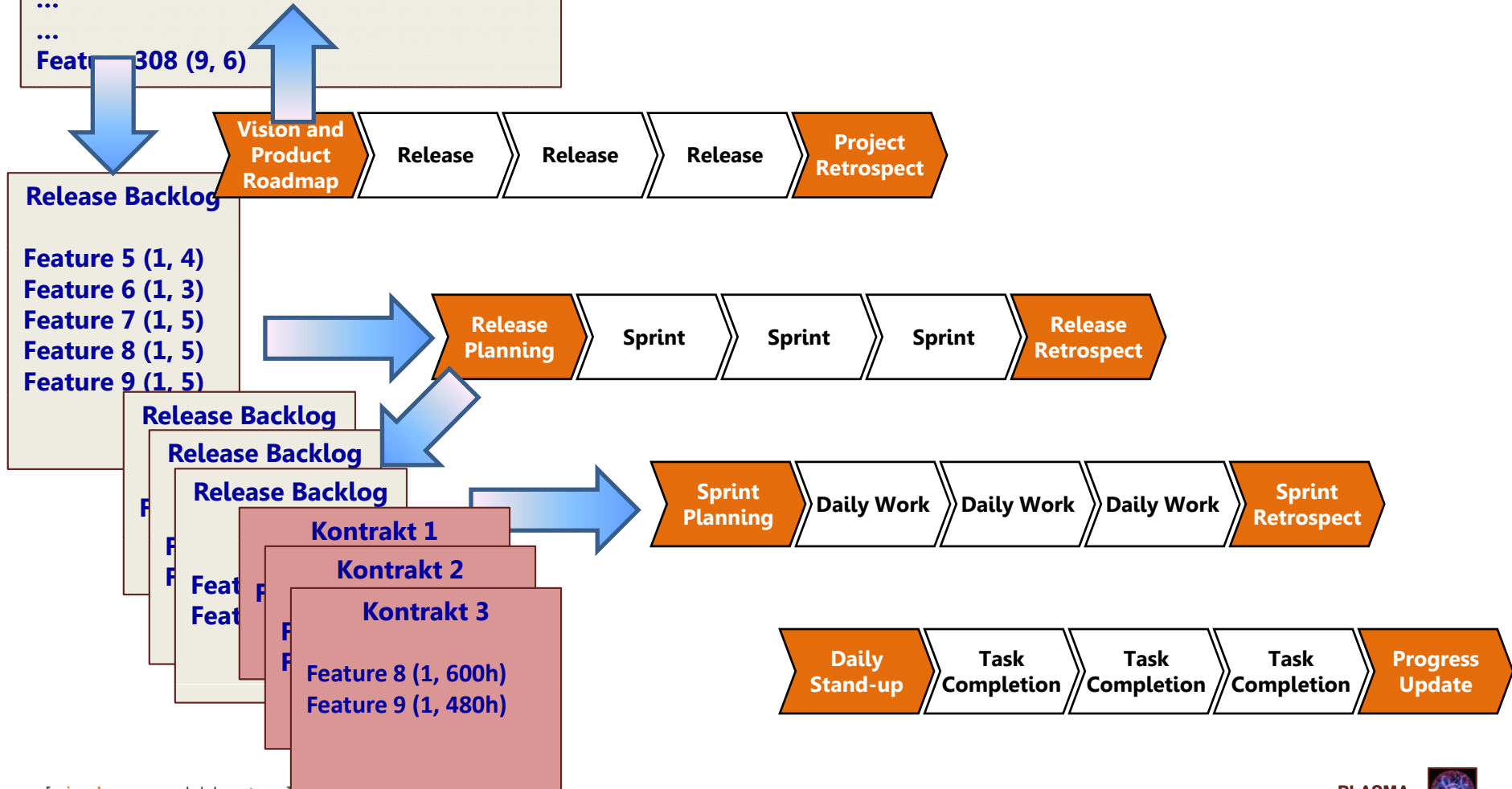




2. Leveransplanering i store smidige prosjekter

Prosjekt PERFORM

80 utviklere fra tre leverandører i 11 Scrum teams, 3 år, NOK 850 millioner



Leveransplanlegging i store smidige prosjekter

Forskningsspørsmål:

Hvordan gjøres leveranseplanlegging i praksis i forhold til hva som antas i planleggingsmodeller og verktøy?

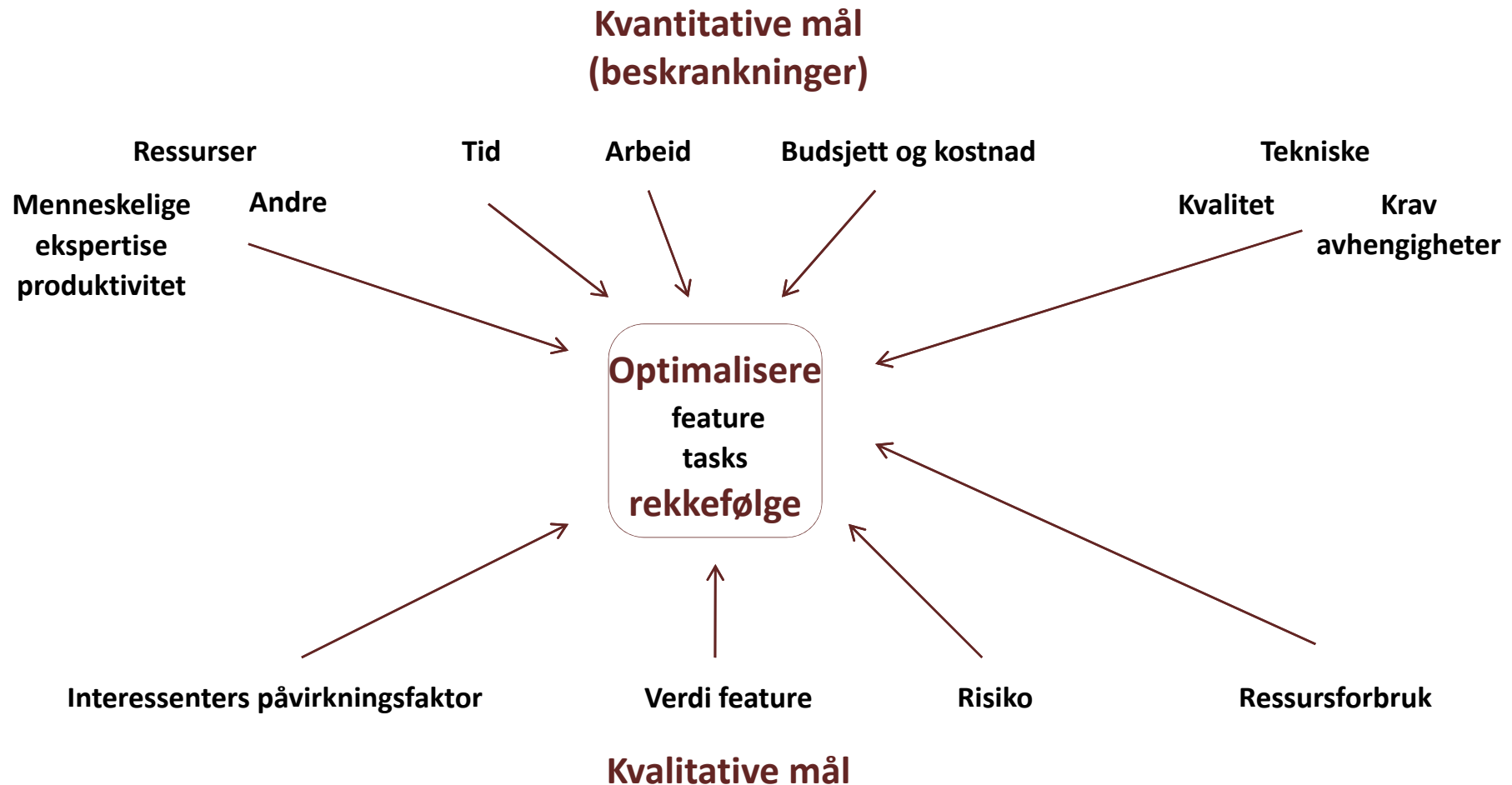


Leveransplanlegging i store smidige prosjekter



Planleggingsmodeller og verktøy

Flere ulike. Oppsummert i Svahnberg et al., A Systematic Review on Strategic Release Planning Models, 2010



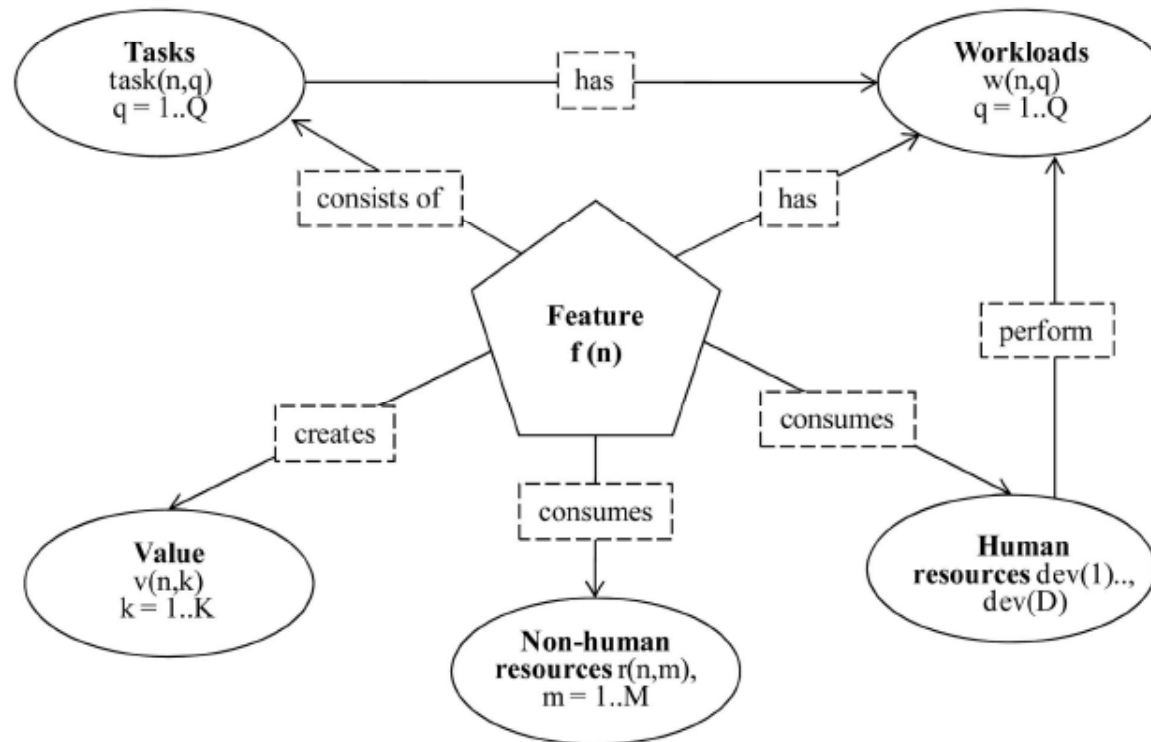
Leveransplanlegging i store smidige prosjekter



Planleggingsmodeller og verktøy

Flere ulike. Oppsummert i Svahnberg et al., A Systematic Review on Strategic Release Planning Models, 2010

Den mest komplette: Ngo-The and Ruhe: Optimized Resource Allocation for Software Release Planning, 2009



Leveransplanlegging i store smidige prosjekter

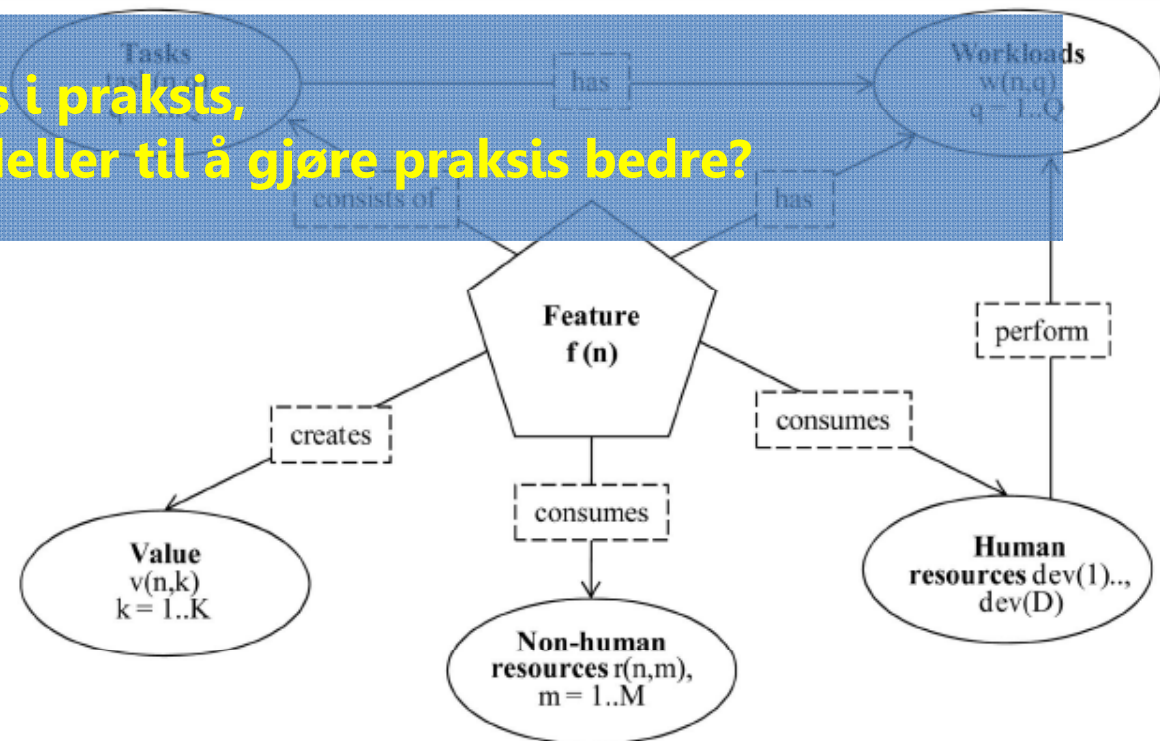


Forskningsspørsmål:

Hvordan gjøres leveranseplanlegging i praksis i forhold til hva som antas i planleggingsmodeller og verktøy?

Modeller: Normative- Slik det bør gjøres i en optimal situasjon

Vi: deskriptive - hva gjøres i praksis, og analytiske - bidrar modeller til å gjøre praksis bedre?



Leveransplanlegging i store smidige prosjekter



Forskningsspørsmål:

Hva gjøres i praksis:

Hvilke faktorer tar eksperter hensyn til ved leveranseplanlegging?

Metode: Mental model elicitation

Repertory grid intervjuer på 5 nøkkelpersjoner i PERFORM

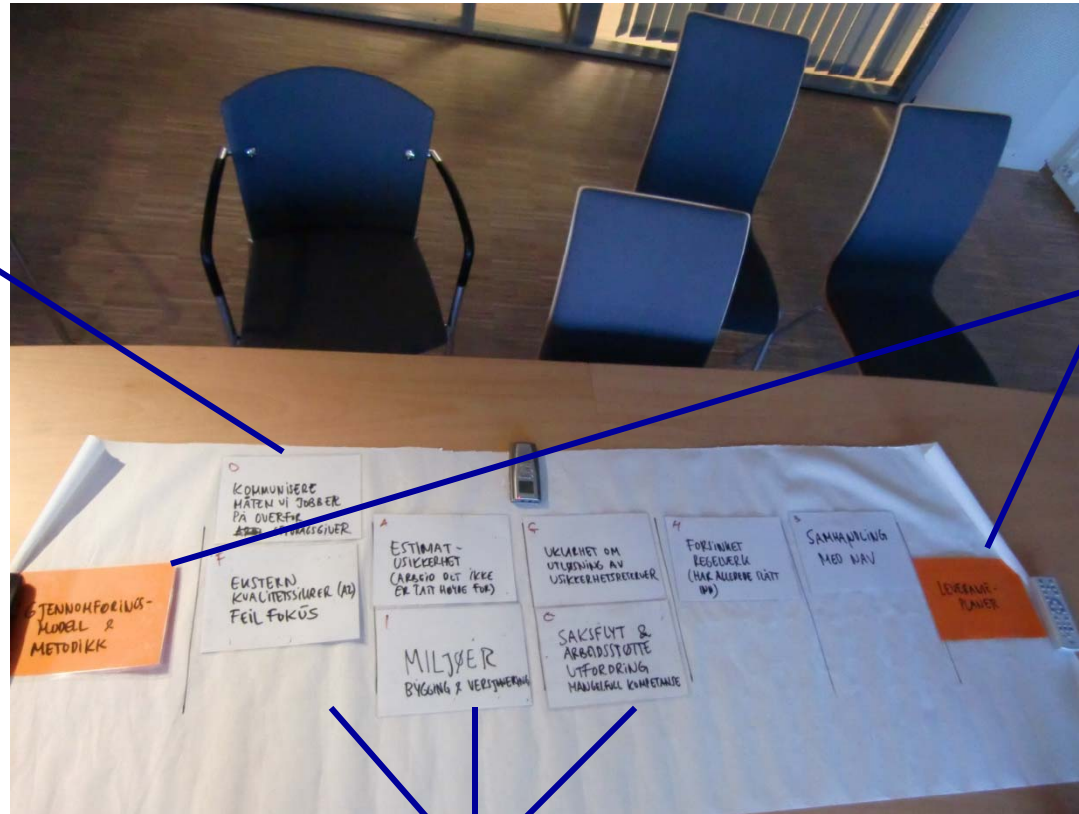
(Kelly: Personal Construct Theory, 1952, Fransella: Repertory Grid Technique, 2004)



Leveransplanlegging i store smidige prosjekter

Repertory grid

Elementer:
Faktorer du tar hensyn til ved leveransplanlegging



Kontraster:
fås ved å sammenlikne tre og tre elementer

Blir dine personlige verdisett

Alle elementene rangeres så på alle kontraster

I tillegg rangeres elementene på

Mindre viktig – Viktig

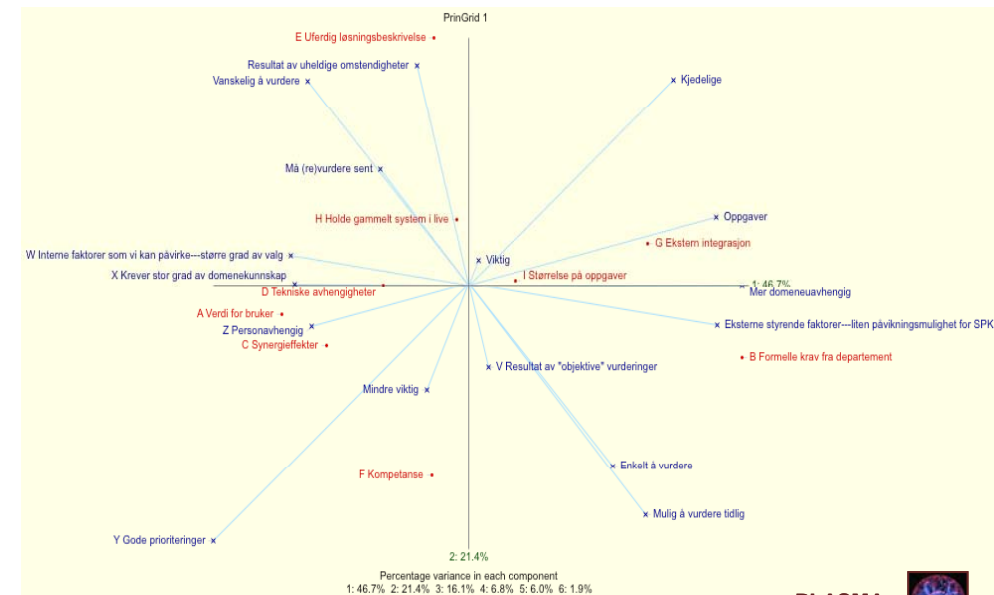
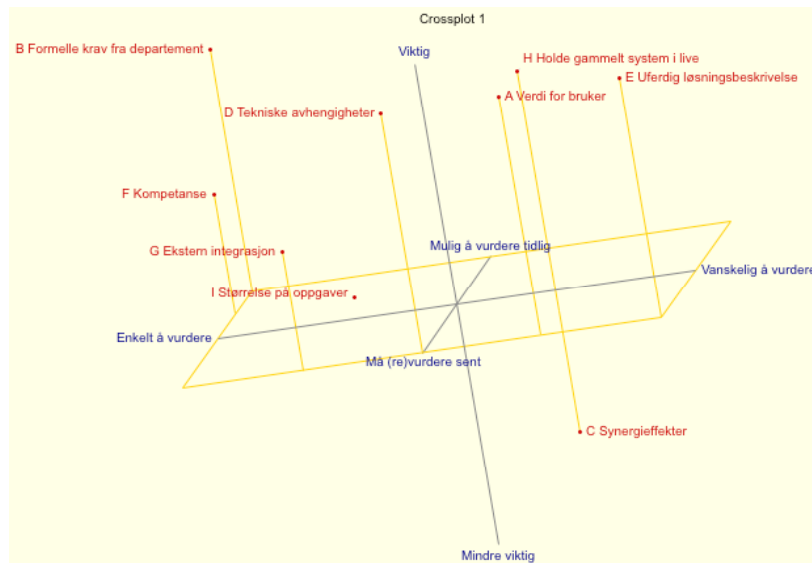
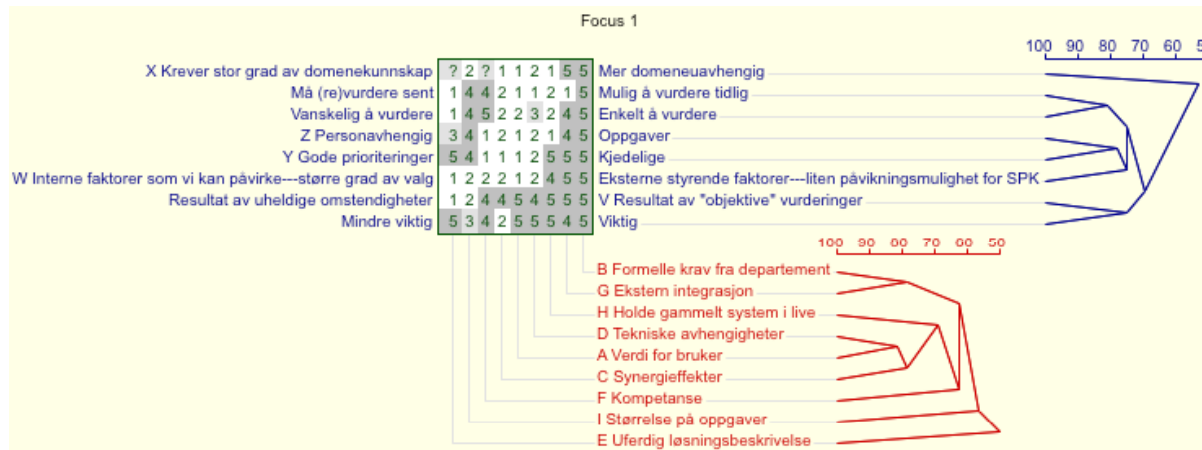
Enkelt å vurdere – Vanskelig å vurdere

Kan vurderes tidlig – Må (re)vurderes sent



Leveransplanlegging i store smidige prosjekter

Repertory grid – Analyser for hvert intervjuobjekt



Leveransplanlegging i store smidige prosjekter

Repertory grid – Oppsummerende analyser



Kategorisere alle elementene (i alt 42) i kategorier.
Resultat: 7 kategorier (hovedfaktorer for leveranseplanlegging)

Kategorisere alle kontrastene (i alt 22) i kategorier.
Resultat: 5 kategorier (hovedkontraster)



Leveransplanlegging i store smidige prosjekter

Svar på forskningsspørsmål:

Hvilke faktorer tar eksperter hensyn til ved leveranseplanlegging?



7 hovedfaktorer for leveranseplanlegging
5 hovedkontraster for leveranseplanlegging



Leveransplanlegging i store smidige prosjekter

De 7 elementkategorier (hovedfaktorer for leveranseplanlegging)

Verdi for bruker

Klarhet og målbildeoppnåelse i løsningsbeskrivelser

Ressurstilgang og roller (kompetanse)

Oppdragsavtale og leveransedato

Avhengigheter i løsning

Teknisk og funksjonell bærekraftighet

Eksterne krav

De 5 kontrastkategorier (hovedkontraster)

Eksterne føringer vs. prosjektets prioriteringer

Hensyn til prosjektorganisering vs. sluttproduktets kvaliteter

Funksjonalitet vs. tekniske kvaliteter

Rammer for leveransen vs. optimal gjennomføring

Dilemma mellom smidig og styring

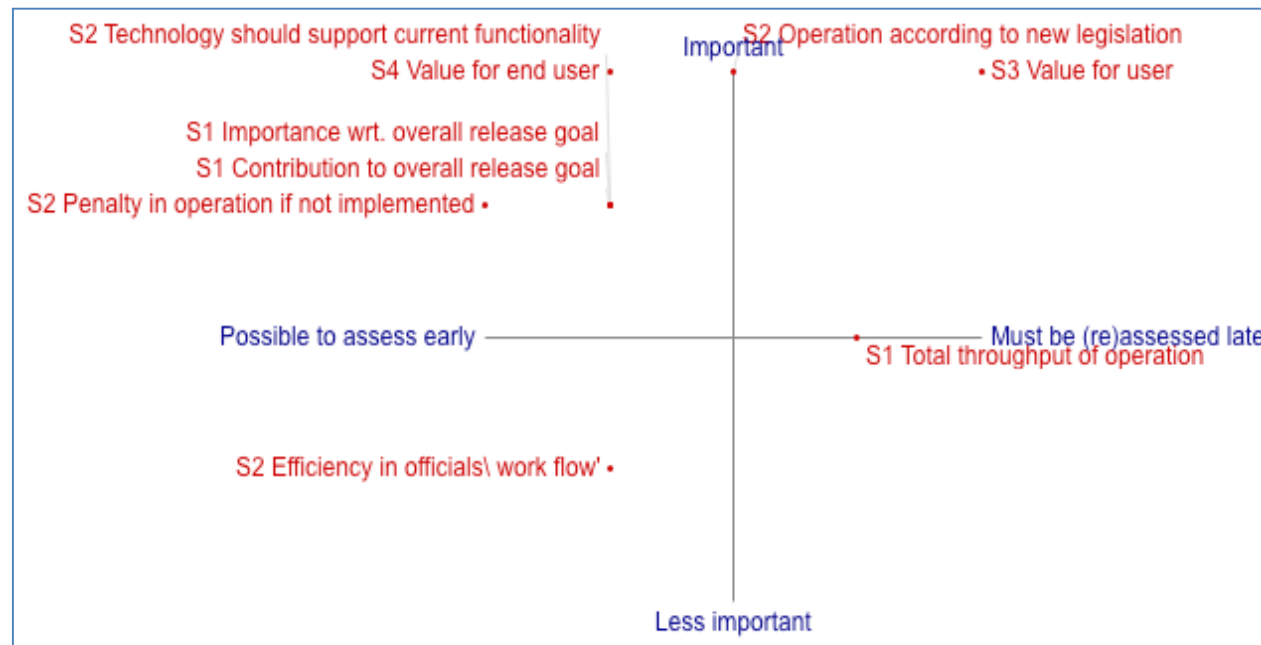


Leveransplanlegging i store smidige prosjekter

Verdi for bruker

- ...er dynamisk over releasene
- ...avhenger av høyere nivå strategisk planlegging

**Verktøy antar at verdi er avklart i starten.
(Kan endres for å teste ulike scenarier)**

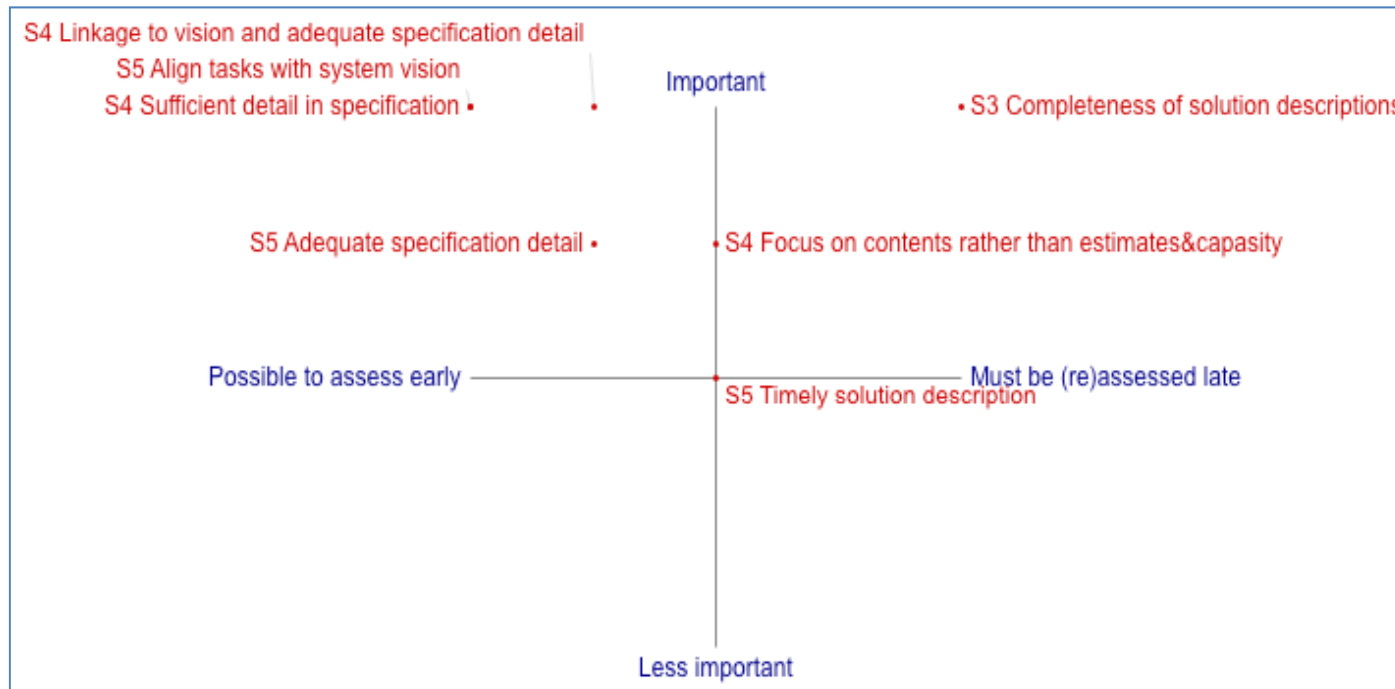


Leveransplanlegging i store smidige prosjekter

Klarhet og målbildeoppnåelse i løsningsbeskrivelser

- ...oppnås gjennom en kontinuerlig kunnskapsdelingsprosess som foregår parallelt med leveranseplanlegging
- ...nødvendig for å definere klart avgrensede og veldefinerte oppgaver

Verktøy krever klart avgrensede og veldefinerte oppgaver

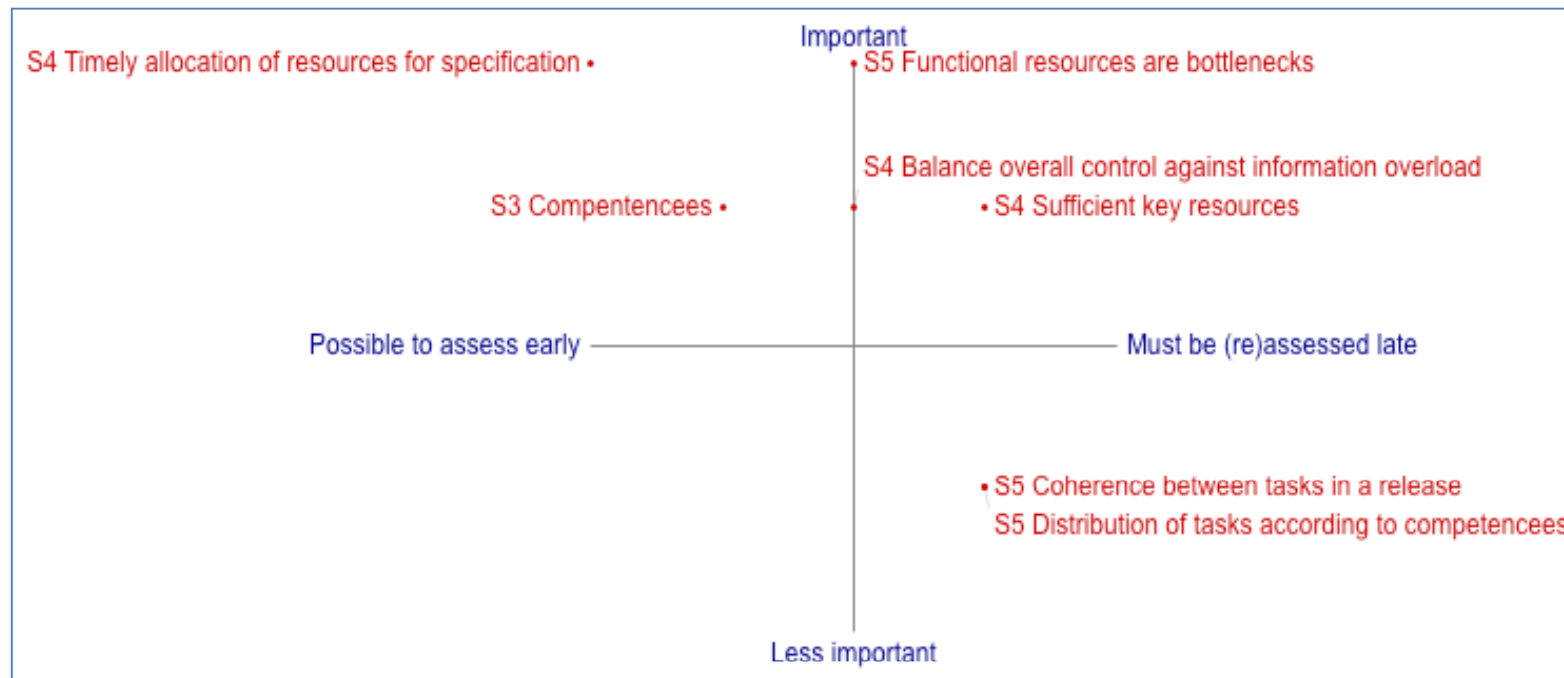


Leveransplanlegging i store smidige prosjekter

Ressurstilgang og roller (kompetanse)

- Funksjonell kunnskap mer kritisk enn utviklingsressurser

Verktøystøttet leveranseplanlegging er bygget opp rundt utviklerressurser

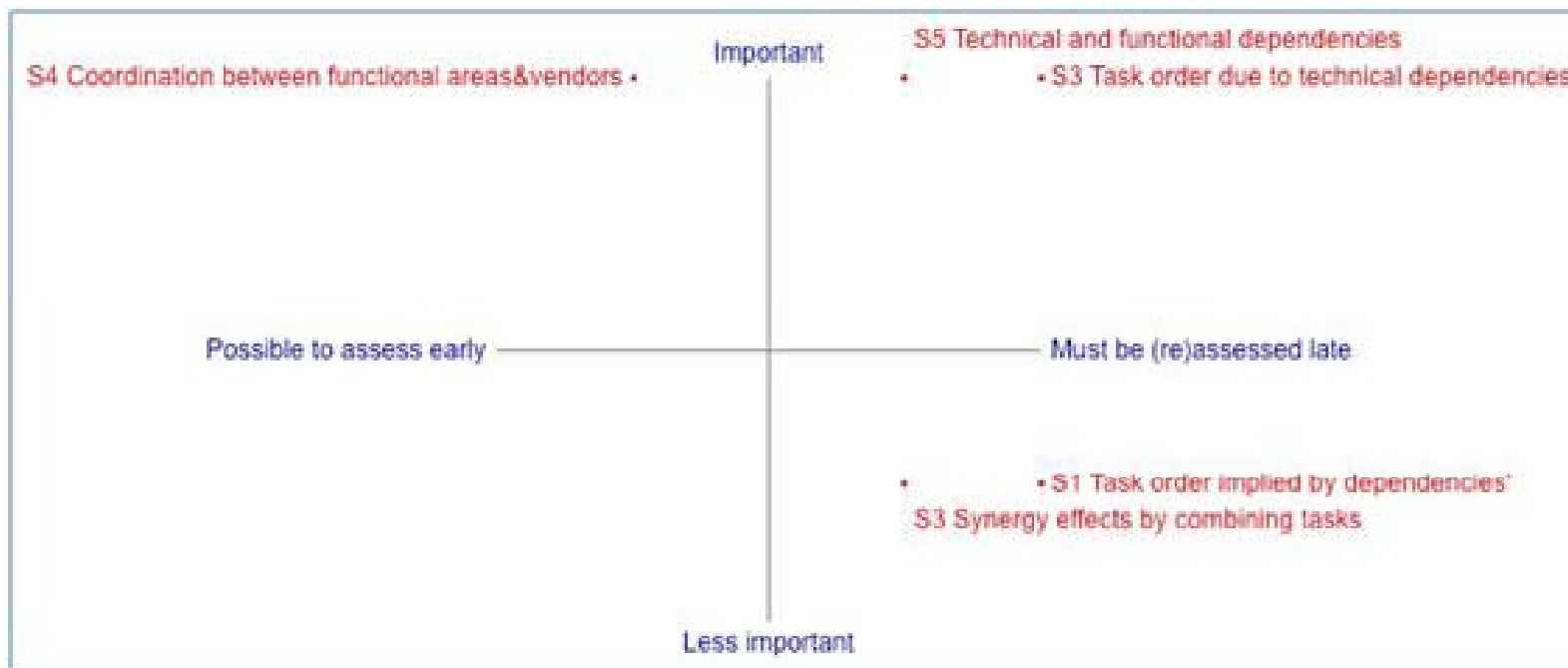


Leveransplanlegging i store smidige prosjekter

Avhengigheter i løsning

- Avhengigheter er helt sentralt i leveranseplanlegging
- ...er vanskelige å avklare (tidlig).

Verktøy krever klare avhengigheter.

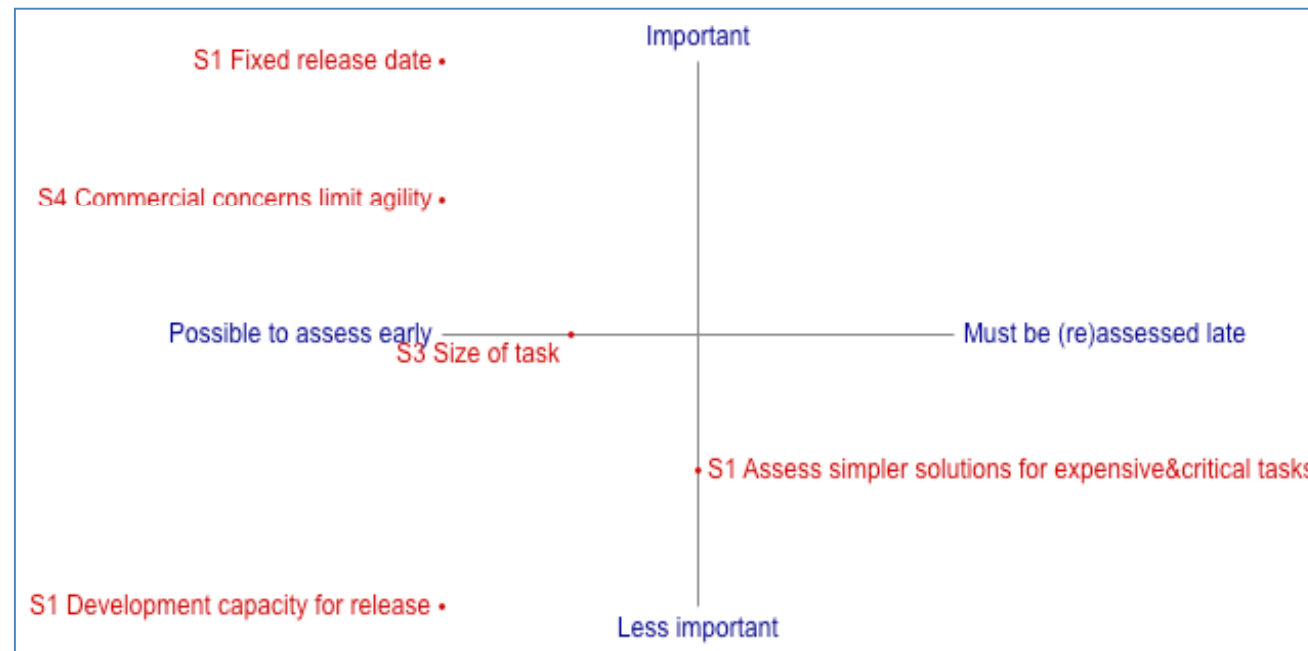


Leveransplanlegging i store smidige prosjekter

Kostnader

- Kritiske features skal prioriteres
- Men kritiske features som er dyre reevalueres for å finne billigere løsninger
- Ledig kapasitet i en release fylles opp med hva det er plass til

Verktøy tar ikke høyde for suboptimale (men pragmatiske) grep.



Leveransplanlegging i store smidige prosjekter

De 7 elementkategorier (hovedfaktorer for leveranseplanlegging)

Verdi for bruker

Klarhet og målbildeoppnåelse i løsningsbeskrivelser

Ressurstilgang og roller (kompetanse)

Oppdragsavtale og leveransedato

Avhengigheter i løsning

Teknisk og funksjonell bærekraftighet

Eksterne krav

De 5 kontrastkategorier (hovedkontraster)

Eksterne føringer vs. prosjektets prioriteringer

Hensyn til prosjektorganisering vs. sluttproduktets kvaliteter

Funksjonalitet vs. tekniske kvaliteter

Rammer for leveransen vs. optimal gjennomføring

Dilemma mellom smidig og styring



Temaer vi ser på:

- 1. Kontraktstandarder i store smidige prosjekter**
- 2. Leveranseplanlegging i store smidige prosjekter**
- 3. Produktivitetshemmere i store smidige prosjekter**



3. Produktivitetshemmere i store smidige prosjekter



Produktivitetshemmere i store smidige prosjekter

Stort og smidig mer og mer vanlig (og nødvendig), men gir motstridende krefter.



Giant scrum?

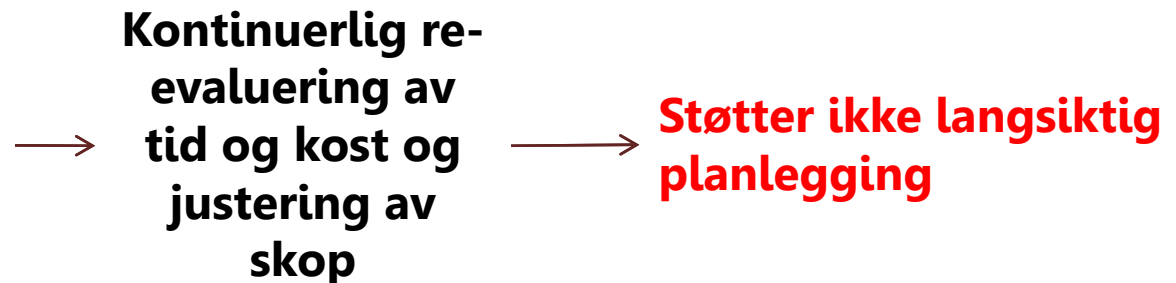
Stort:

- Mye penger
- Ekstrem kompleksitet
- Høy risiko



Smidig:

- Krav som kommer sent eller endres
- Ingen unødvendig funksjonalitet



Mange anekdoter og anbefalinger for stort + smidig. Lite forskning.



Produktivitetshemmere i store smidige prosjekter

PERFORM – et stort prosjekt som anvender smidig utviklingsmetodikk



PERFORM er et critical case: Prosjektet er heldig stilt gjennom dyktige medarbeidere i nøkkelroller, god forankring av smidig prosess i alle organisasjonsledd, og stor grad av brukermedvirkning.

Hvis man identifiserer utfordringer under disse gunstige forhold, kan man forvente at liknende utfordringer oppstår i andre mindre heldigstilte prosjekter.



Produktivitetshemmere i store smidige prosjekter



Forskningsspørsmål:

Hva gjøres i praksis:

Hvilke hindere til produktivitet opplever du i prosjektet?

Metode: Mental model elicitation

Repertory grid intervjuer på 13 personer (tverrsnitt) i PERFORM

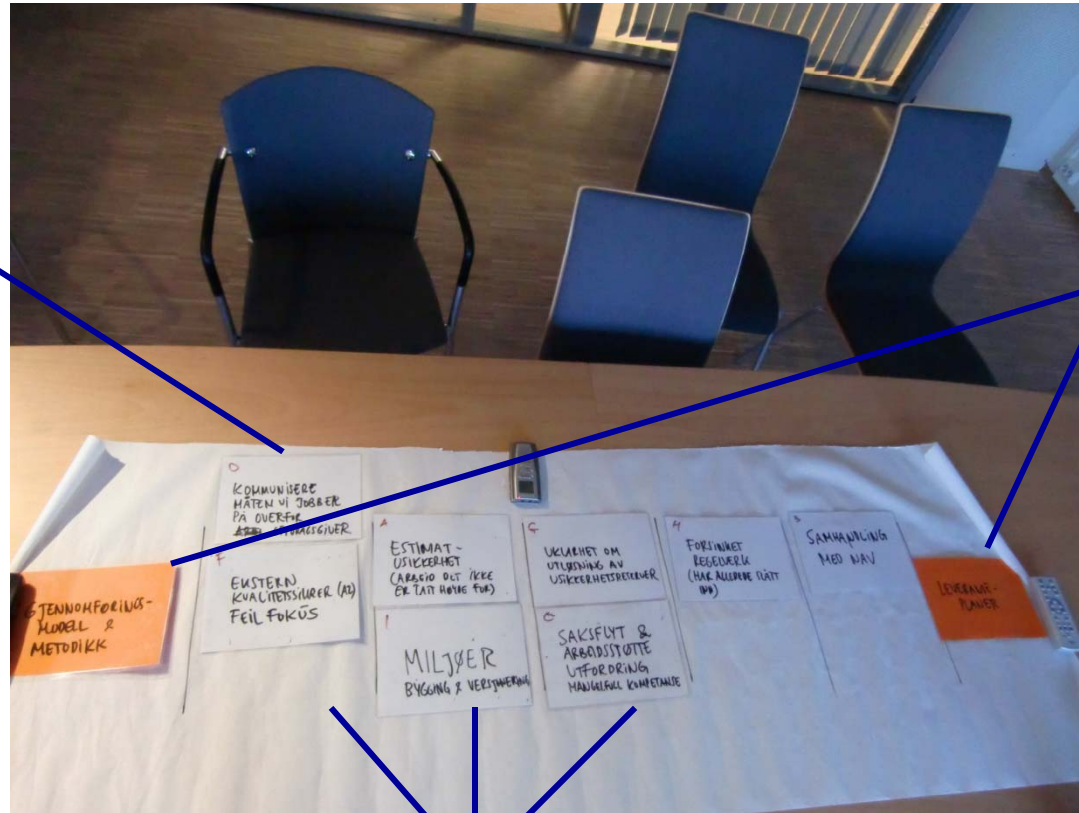
(Hannay & benestad ESEM2009, Kelly: Personal Construct Theory, 1952, Fransella: Repertory Grid Technique, 2004)



Produktivitetshemmere i store smidige prosjekter

Repertory grid

Elementer:
Hindre for optimal produktivitet



Kontraster:
fås ved å sammenlikne tre og tre elementer

Blir dine personlige verdisett

Alle elementene rangeres så på alle kontraster

I tillegg rangeres elementene på

Ikke så farlig – Alvorlig

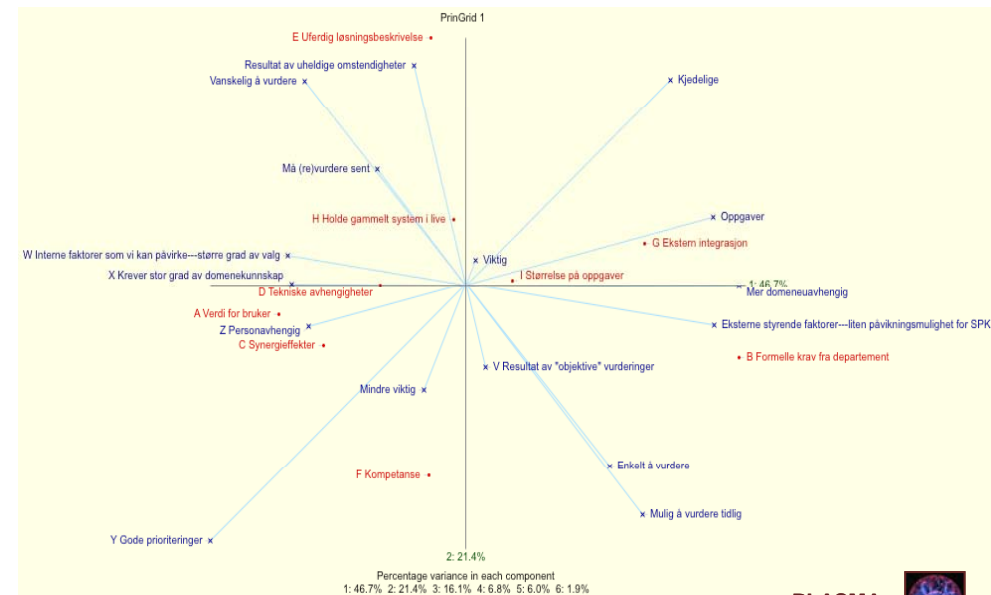
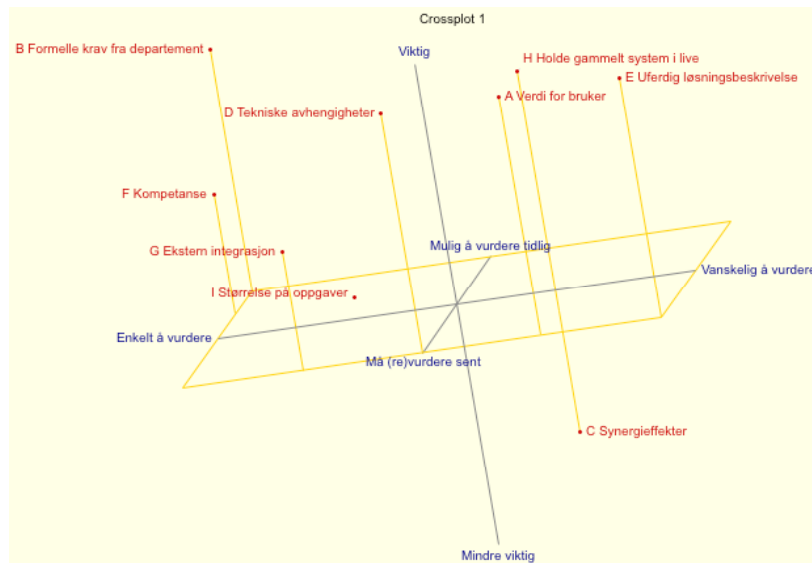
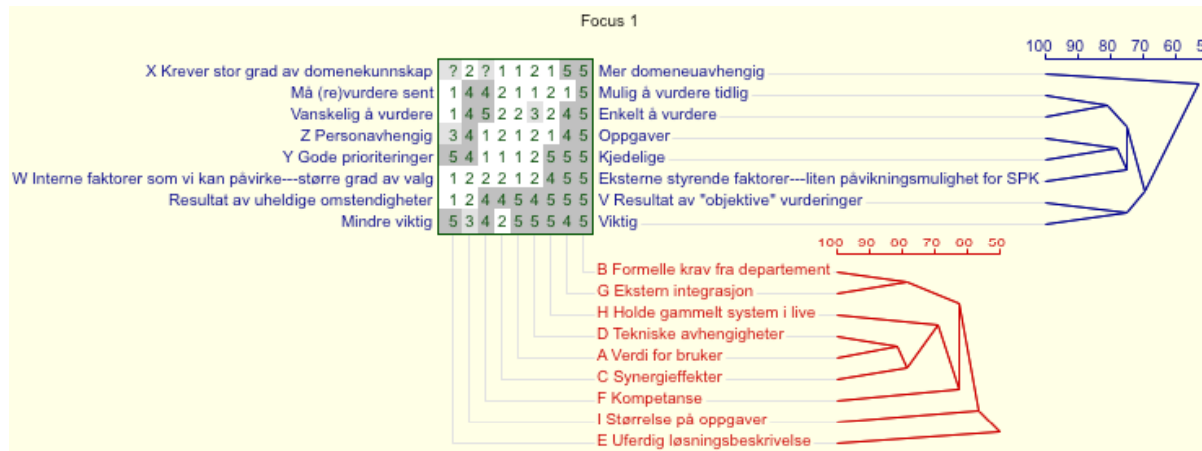
Enkelt å håndtere – Vanskelig å håndtere

Årsak - Virkning



Produktivitetshemmere i store smidige prosjekter

Repertory grid – Analyser for hvert intervjuobjekt



Produktivitetshemmere i store smidige prosjekter

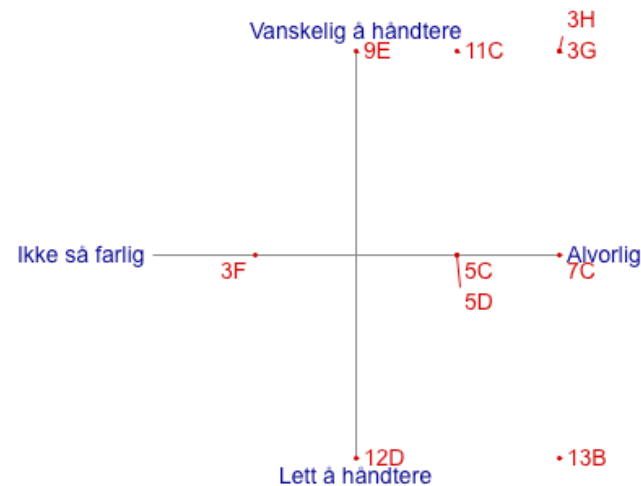
Repertory grid – Oppsummerende analyser



Kategorisere alle elementene (i alt 100) i kategorier.
Resultat: **10 kategorier (hovedproblemstillinger)**

Kategorisere alle kontrastene (i alt 22) i kategorier.
Resultat: **5 kategorier (hovedkontraster)**

Oppsummere **alvorlighetsgrad**
og **aksjonsmulighet** for
elementer i hver kategori



Produktivitetshemmere i store smidige prosjekter

Svar på forskningsspørsmål:

Hvilke hindere til produktivitet opplever du i prosjektet?



10 hovedproblemområder



Produktivitetshemmere i store smidige prosjekter

Vi spurte 13 prosjektdeltakere om deres oppfatninger av produktivitetshemmere i PERFORM



10 problemområder:

1. Kontraktsmessige forhold og kulturforskjeller begrenser fellesskapstenking og prosessforbedring
2. Lav prioritet til arkitektur og tekniske kvaliteter forsinker ferdigstilling av produksjonsklar kode
3. Ubalanse mellom hensynene til styring og fleksibilitet gir sub-optimale prosesser for prosjektstyring, utvikling og test
4. Endringer og forsinkelser i eksterne føringer belaster nøkkelressurser og gir overskridelser i estimater
5. Mangel på felles visjon for systemet hindrer utvikling av deløsninger som drar i samme retning
6. Begrenset spredning av rik funksjonell kunnskap er til hinder for effektiv utvikling og gode løsninger
7. Koordinering av avhengigheter mellom ulike deler av produktet er utfordrende og mangelfull
8. Lav tilgjengelighet på nøkkelkompetanse gir overbelastning hos nøkkelpersoner og hindrer utvikling av god løsning
9. Konfigurering av effektive og behovstilpassede tekniske miljøer for utvikling og test er vanskelig og tidkrevende
10. Koordinering av utvikling, test og produksjonssetting mot linje og eksterne parter er vanskelig og tidkrevende



Kontraktsmessige forhold og kulturforskjeller begrenser fellesskapstenking og prosessforbedring

Diverse aktører har egne interesser å ivareta

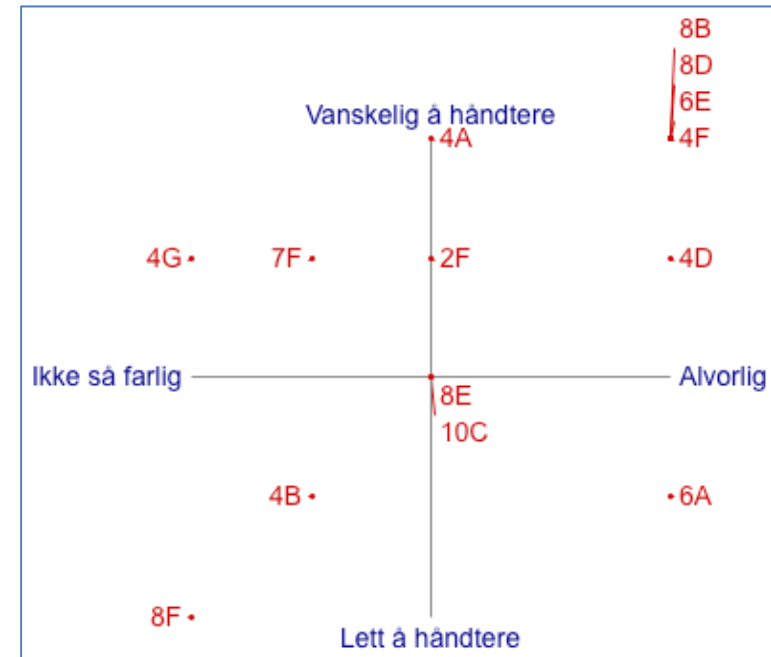


Mye av dette er uttalt og implisitt!

Egne interesser:

- kontrakt-/anbudsmessige forhold
- økonomi og eierskap
- personlige syn på løsning og prosess

Konsekvenser er at man får ufullstendig team-kompetanse, at team opplever lediggang, og at kommunikasjon mellom teamene hemmes.



Mulige startpunkter for forbedring:

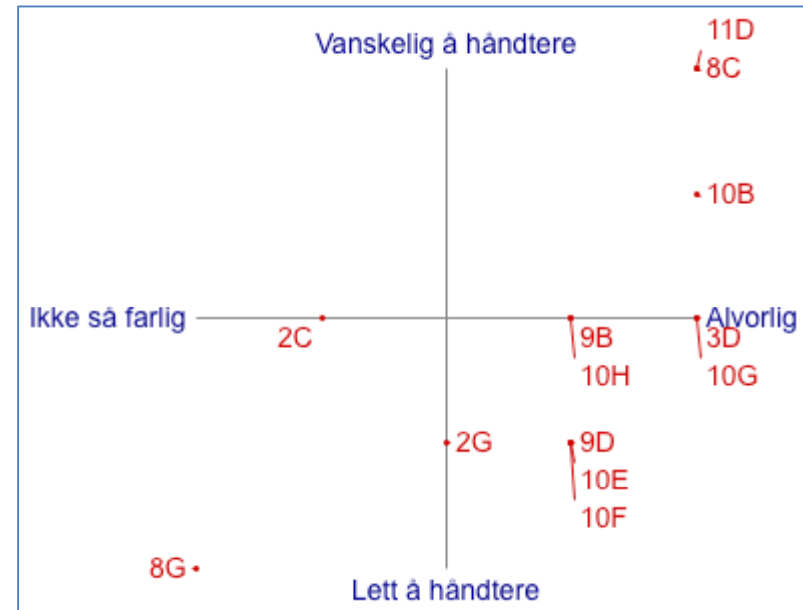
1. Kjø for "overskuddsoppgaver" (feks tekniske kvaliteter), og incentiver for å løse disse
2. Fast rutine for å vurdere spesielle kompetansebehov før hver iterasjon



Lav prioritet til arkitektur og tekniske kvaliteter forsinker ferdigstilling av produksjonsklar kode



Teamene møter problemer (ytelse, driftbarhet, testbarhet, unødvendige avhengigheter) som kunne ha vært unngått med et høyere fokus på arkitektur og tekniske kvaliteter, både i starten av prosjektet og under løpende prioritering i iterasjonene



Mulige startpunkter for forbedring:

1. Kjø for forbedring av tekniske kvaliteter (med incentiver)
2. Ett mulig kjønnslag: Task force for optimal bruk av FitNesse
3. Mer systematisk test av tekniske kvaliteter ved kontrollpunktene



Ubalanse mellom hensynene til styring og fleksibilitet gir sub-optimale prosesser for prosjektstyring, utvikling og test

Felles produktkø med tidlige og bindende estimater

Eierskap til del-løsninger

Godkjenningstest

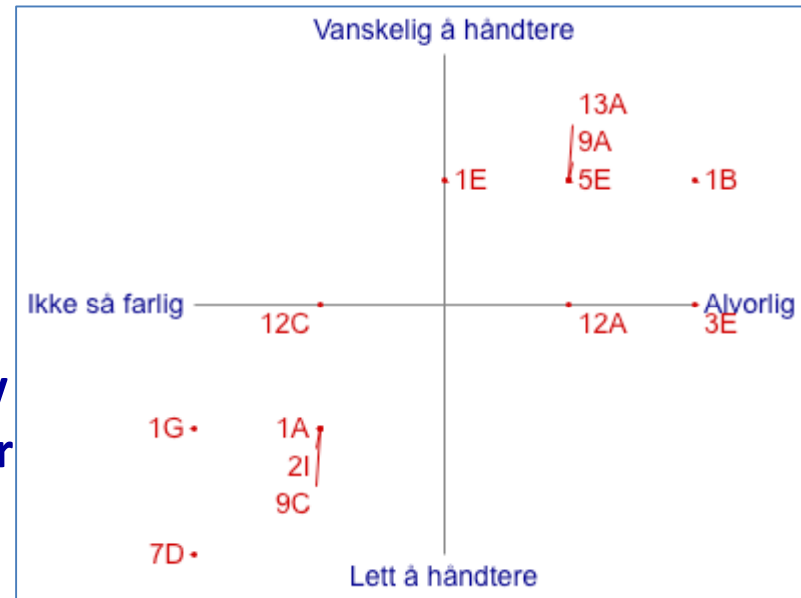


Løpende behovsanalyse og estimering

Jevn fordeling av mindre oppgaver

Kontrollpunkt-test

Mangel på fleksibilitet hindrer motivasjon og en mer fornuftig prioritering av oppgaver. Det store fokus på godkjenningstest går ut over kvaliteten på kontrollpunkt-testene.



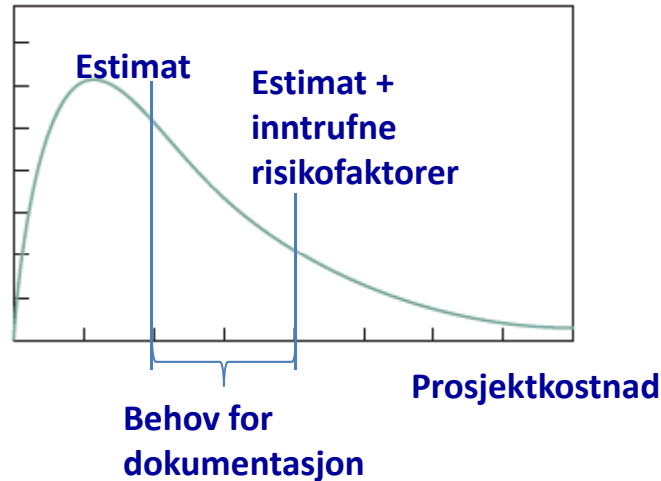
Mulige startpunkter for forbedring:

1. Kontinuerlig vurdere om elementene i masterplan er nødvendige og tilstrekkelige
2. Mer fleksibilitet i prioritering av oppgaver – se estimater for et større antall oppgaver i sammenheng
3. Forskyve testressurser fra godkjenning til kontrollpunkter

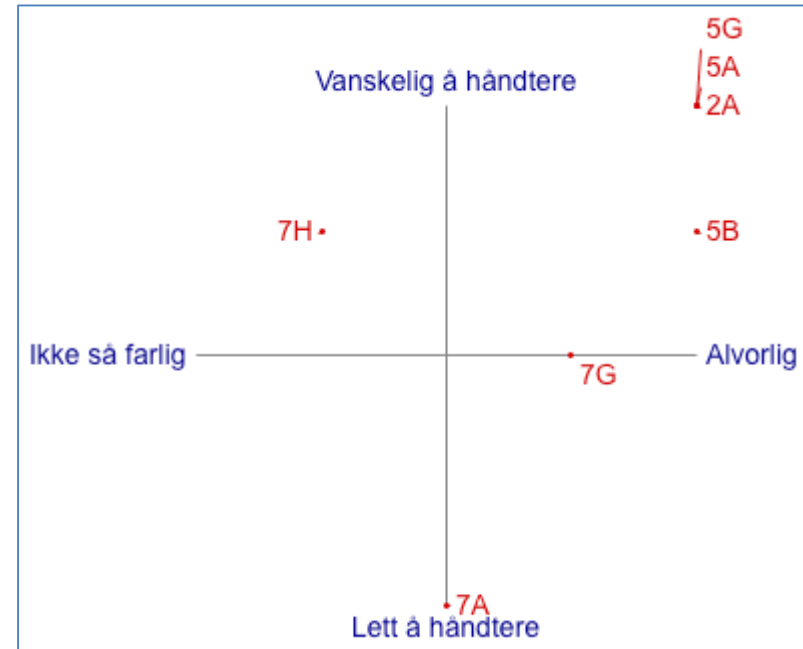


Endringer og forsinkelser i eksterne føringer belaster nøkkelressurser og gir overskridelser i estimater

Sannsynlighet



Kjente (sent regelverk) og ukjente risikofaktorer har inntruffet og forsinker prosjektet i forhold til Estimat. Selv om disse faktorene er dokumenterte, utløses ikke nødvendigvis reservemidler. Må i stedet gjennom ressurskrevende QA kontroll.



Mulige startpunkter for forbedring:

1. Avklare dokumentasjonskrav for risiko/usikkerhet og formidle kvalitet av estimater og usikkerhetsanalyse
2. Vurdere tidshorisonter og detaljnivå for estimering og re-estimering
3. Ha løsningskompetanse ved estimering



Mangel på felles visjon for systemet hindrer utvikling av delløsninger som drar i samme retning

Vi har et mål-bilde, men det er ikke godt nok kommunisert



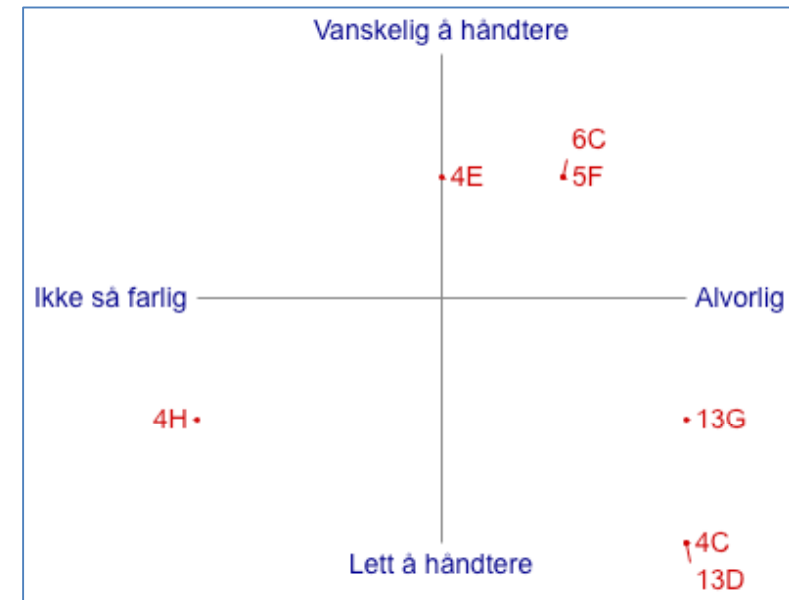
Vi er usikre på om det gode overordnede oversikten og målbildet faktisk finnes

Nyansforskjeller i hvordan dette oppfattes

Uansett nyanse oppfattes dette å påvirke prosjekts suksess gjennom:

Løsningens behovsoppfyllelse

Dobbeltarbeid/tilbakekalling av kode



Mulige startpunkter for forbedring:

1. Sette av tid til å kommunisere målbildet til hverandre
2. Fjerne usikkerhet om hvorvidt total oversikt finnes



Begrenset spredning av rik funksjonell kunnskap er til hinder for effektiv utvikling og gode løsninger

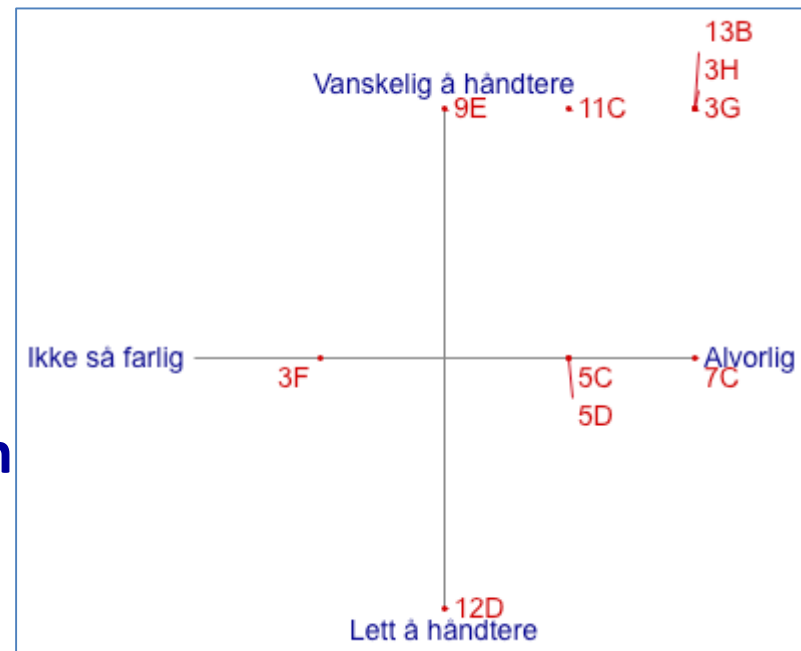
Kunnskap om gammelt system

Kunnskap om brukere

Kunnskap om domenet



Kunnskap om NAV integrasjon



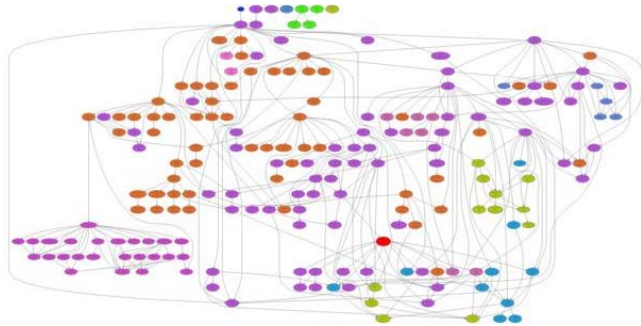
Mulige startpunkter for forbedring:

1. Allokere hensiktsmessig funksjonell kompetanse på kritiske steder
2. Koordinere tidspunkter for nødvendig infolyt fra forretning til utvikling
3. Hjelp til å overføre funksjonell kunnskap ved testspesifisering

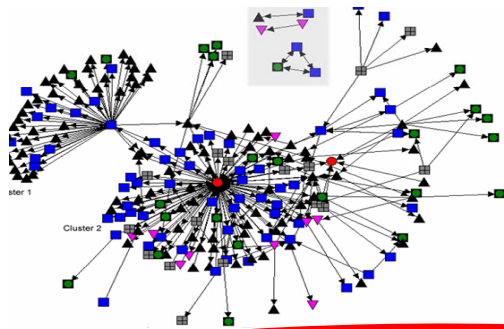


Koordinering av avhengigheter mellom ulike deler av produktet er utfordrende og mangelfull

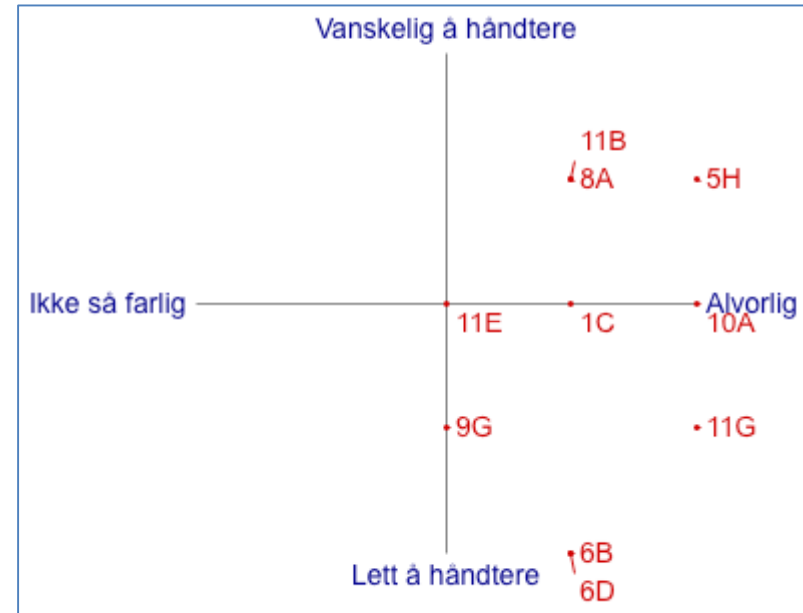
Produktavhengigheter



Koordineringsbehov



Mange avhengigheter å kommunisere om, og kommunikasjonen om dette flyter ikke lett mellom teamene. Dermed mye tid brukt på endringer i etterkant som følge av utilstrekkelig koordinering. Har hendt at to team har utviklet det samme.



Mulige startpunkter for forbedring:

1. Identifisere overlappende kjø- elementer
2. Allokere ansvar (mer eksplisitt) for koordinering av testdata
3. Avklare avhengigheter tidligere
4. Vurdere hvor mye nettforståelse forskjellige deler av prosjektet ska ha



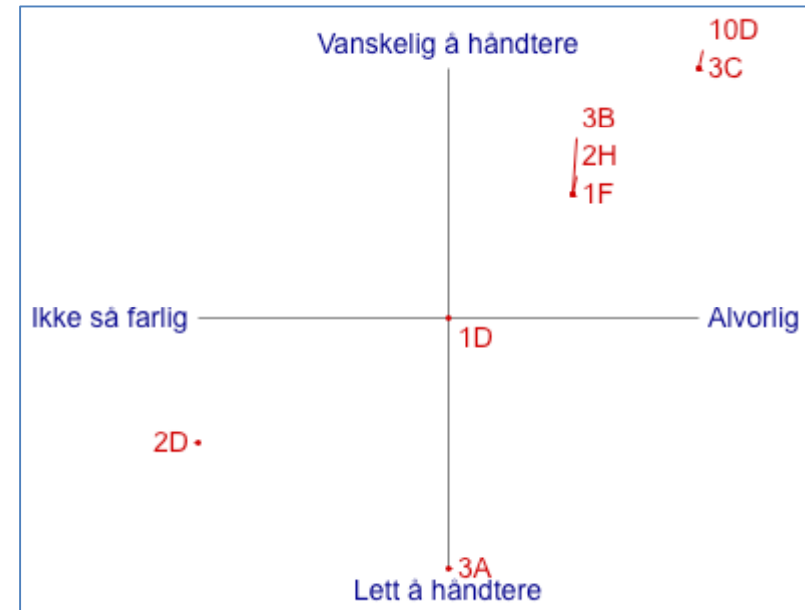
Lav tilgjengelighet på nøkkelkompetanse gir overbelastning hos nøkkelpersoner og hindrer utvikling av god løsning



Nøkkelpersoner blir flaskehals---spesielt personer med kunnskap både om forretningen og løsningen

Personer har mange samtidige roller, pga delprosjekter som krever kompetanse på tvers av prosjektorganisasjonen

Kontekstswitching er tidkrevende og produktivitetshemmende

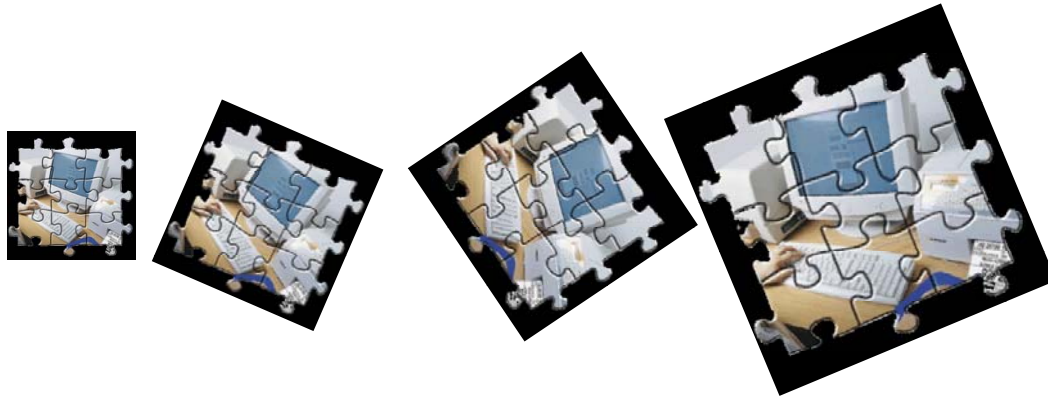


Mulige startpunkter for forbedring:

1. Eksplisitt avstemme rollefordeling og rolleforventning
2. Kan nøkkelpersoners tilgjengelighet gjøres forutsigbar?
3. Redusere utskiftninger av nøkkelpersonell i teamene

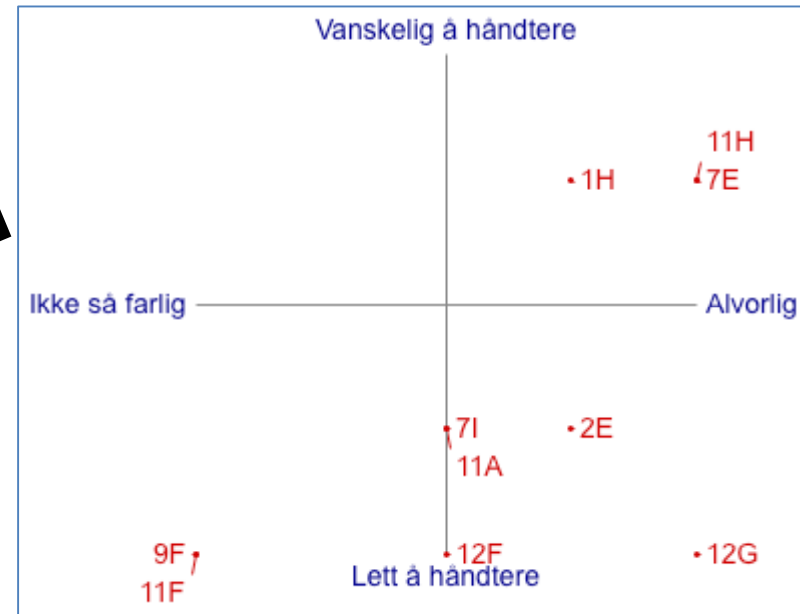


Konfigurering av effektive og behovstilpassede tekniske miljøer for utvikling og test er vanskelig og tidkrevende



Kontinuerlig behov for endringer i oppsett

Mange varianter i parallell og behov for stadig endring av utviklings- og testmiljø krever mye ressurser. Miljøene fungerer ikke alltid godt nok og hemmer og forsinker teamene. Vanskelig tilgjengelig også for ikke-teknisk personell.

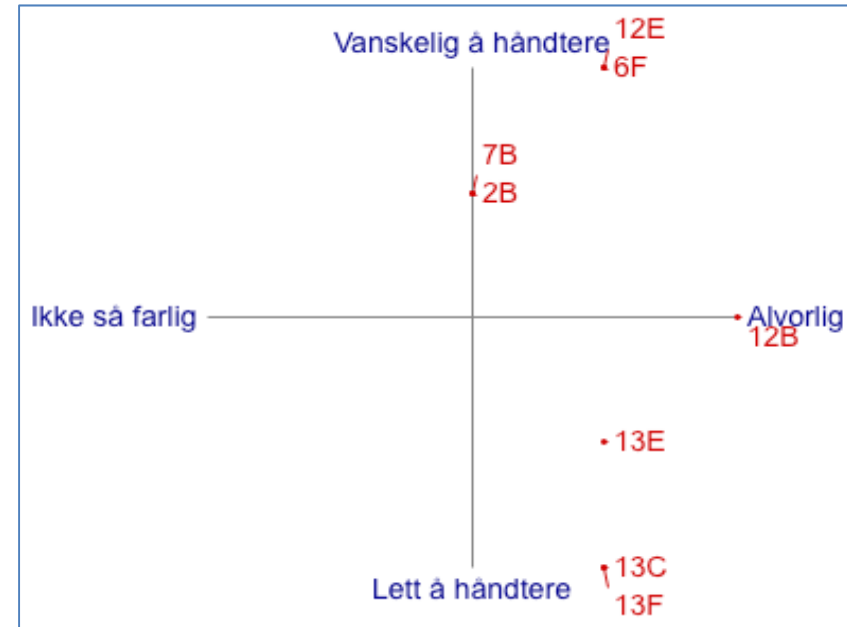
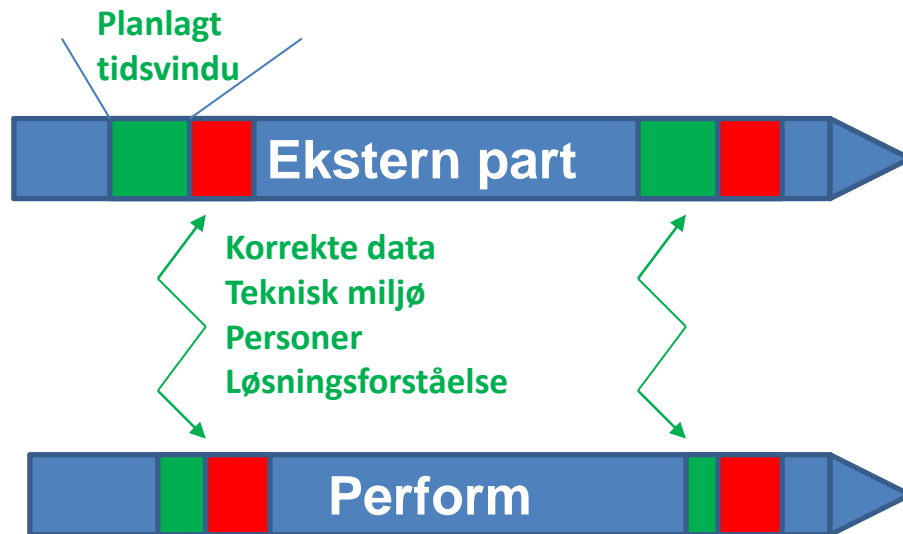


Mulige startpunkter for forbedring:

1. Verktøy for å finne ut hvilke områder som inneholder mye feil
2. Hva skal til for å sette opp testmiljøet raskere for godkjeningsprøven?
3. Revurdere bruken av teknologi i utviklingsmiljøet



Koordinering av utvikling, test og produksjonssetting mot linje og eksterne parter er vanskelig og tidkrevende



Det brukes mye ressurser på å tilrettelegge for tester og produksjonssetting hos eksterne parter (inkludert SPK sin linje). Liten kalendertid er satt av, og det blir ekstra støy og ressursforbruk når tidsvindu sprekker. .

Mulige startpunkter for forbedring:

1. Finne og binde de rette kontaktpersoner hos eksterne parter
2. Ha brukerdokumentasjon på et nivå som linja er fortrolig med



Mette...

