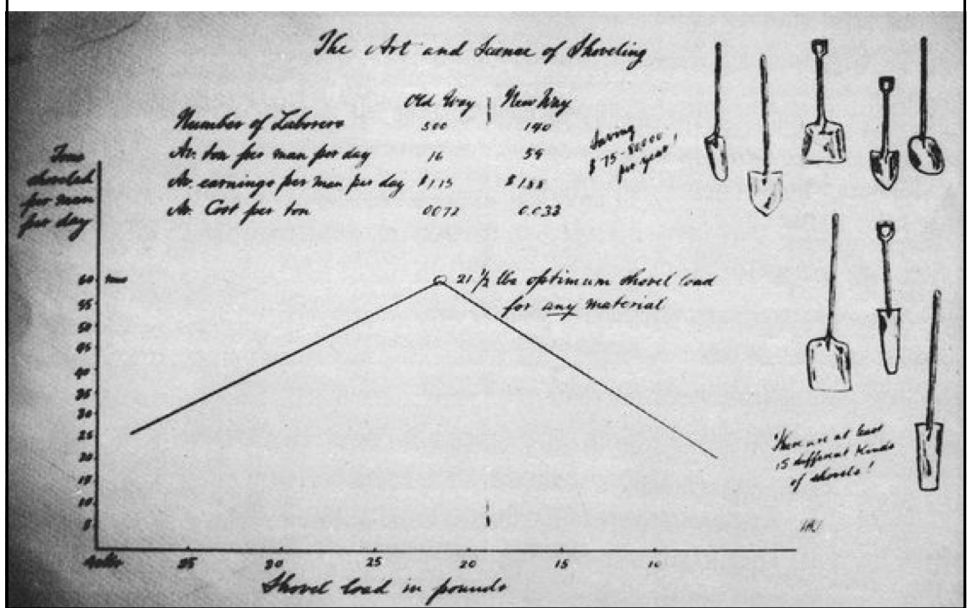




Evidensbaserte arbeidsprosesser og verktøy ...
Vet dere hva dette er?



En liten undersøkelse

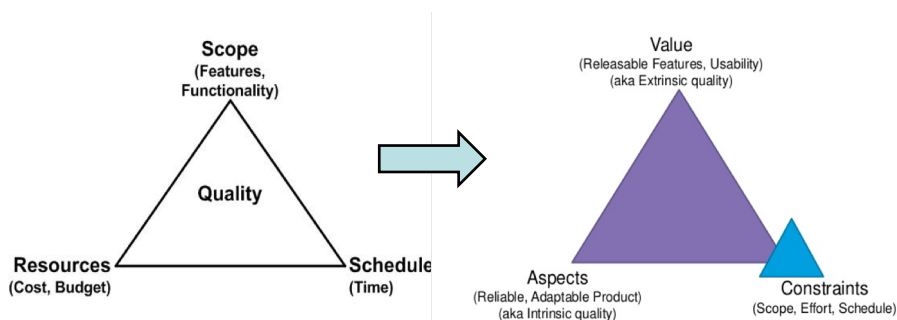
(Tar: 10-15 minutter)

tinyurl.com/sivert2019

Resultatene, og hva dere kan lære, vil bli oppsummert i løpet av presentasjonen.



Hva vil det si å lykkes?



Suksess er kontekststøttet, men ofte en prioritert kombinasjon av:

- Oppnådd nytte (gevinster, måloppnåelse, ROI)
- Tekniske egenskaper (kvalitet, utvidbarhet)
- Prosjekteffektivitet
- Tidskontroll
- Kostnadskontroll

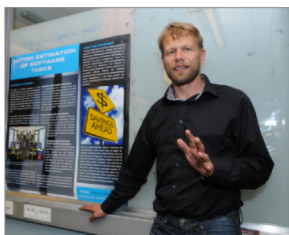
De fleste prosjekter lykkes rimelig godt ...

- **Gjennomgående resultat fra flere ulike, norske spørreundersøkelser de siste fem årene:**
 - Ca. 50% av prosjektene blir evaluert som "akseptable" eller bedre på suksesskriteriene (nytte, funksjonalitet, kvalitet, kostnad, tid og effektivitet)
 - Ca. 50% blir evaluert som mislykket på minst ett av kriteriene
 - Ca. 10% feiler helt (blir enten stoppet, eller leverer svært lite nytte)

Offentlig sektor synes å ha blitt flinkere:

Computerworld: 6. juni, 2016

2004: Betydelig hyppigere med problematiske IT-prosjekter i offentlig enn i privat sektor



2015: Offentlig sektor og privat sektor lykkes ca. like godt med IT-prosjekter.

- Offentlig sektor har blitt **flinkere**

Det sier professor Magne Jørgensen. Men virksomheter sliter fortsatt med gevinstene på sikt



Jobb og ledelse Arbeidsliv

Flere fiaskoer med fastpris

Hvordan konsulentene betales, kan avgjøre om et it-prosjekt blir fiasko eller suksess, ifølge ny forskning på flere hundre norske it-prosjekter.



DET ER BLITT GJORT MANGE FORSØK på å finne felles trekk ved mislykkede offentlige IT-prosjekter. Store, ambisiøse, langvarige prosjekter med ferdige kravspesifikasjoner er ett problem. Skreddersøm er et annet. Simulas forskning bidrar med nye funn.

«Det kanskje sterkeste mønsteret av alle starter med valg av kontraktstype», sa sjefforsker Magne Jørgensen i Simula til Dagens Næringsliv mandag. Prosjekter der IT-leverandøren får timepris i stedet for en fast pris for hele jobben, lykkes oftere. Prosjekter i Statens pensjonskasse, Digipost og Lånekassen er alle eksempler på dette.

Andre anbefalinger i Simula-rapporten er å fokusere på nytte, ikke kravspesifikasjoner, samt å legge opp til smidige prosesser med hyppige leveranser som evalueres av kunde underveis, bare for å nevne noe.

IKT-FIASKO I DET OFFENTLIGE:

Halvparten av alle norske IKT-prosjekter havner i problemer



Store prosjekter har større risiko for å mislykkes enn små prosjekter.



Prinsipper for redusert risiko og større gevinster

1. Start med behov
2. Tenk stort - start smått
3. Velg riktig samarbeids-partner
4. Riktig kompetanse og god lederforståelse
5. Lever hyppig - skap nytte hele veien

Nå orker vi ikke flere IT-satsinger over statsbudsjettet. Hilsen en som tjener på IT-satsinger. | Aslak Ege

Hva tilsier forskningen vår?

Lite representativ for Norge

Pengsluket er i prosjektstyringen

En undersøkelse fra McKinsey/Oxford University (2012) viser at rundt to av tre store IT-prosjekter sprenger tiden eller budsjettet. Og her er noen av årsakene de lister opp:

- Manglende evne til å lede strategi og interessenter
- Lav forståelse av innhold og teknologi
- Manglende forståelse for prosjektledelse

Undersøkelsen sier ikke noe spesifikt om den norske staten, men den sier oss at pengsluket ligger i prosjektstyringen. Og så lenge lav forståelse for teknologi er en faktor, bør det fortelle oss hva byråkrater skal gjøre og hva teknologer skal gjøre.

Feil

Helt motsatt!!!

Ikke hva som skjer

Hva menes? Finnes løsninger som ikke må videre-utvikles?

Probleme kommer når staten skal blande seg inn i hvordan noe skal gjøres.

La meg likevel bringe litt håp inn i dette. Her er mine tre håp for neste statsbudsjett:

At staten nøyer seg med å eie, ikke lede, IT-prosjektene. Stol på at kompetente IT-folk gjør gode valg. Det sparer skattebetalerne på.

At staten slutter å kaste gode penger etter dårlige penger. Man må slutte å tro at man kjøper rimelig bare fordi utviklingsfasen er billig. For det kommer nemlig en driftsfase. Da skal det forvaltes. Det koster penger.

At staten kjøper ordentlige løsninger som spiller av seg selv. Det er litt dyrere å utvikle sånt som går av seg selv. Men etterpå slipper horder av konsulenter å forvalte løsningene. Altså løpe ved siden av i ti år og fikse bugs eller dårlig kode sånn at vi plutselig må ha et krafttak igjen. Men jeg er også realist, så jeg er forberedt på at disse håpene blir knust i statsbudsjettet. For da skal det nok satses igjen.

Hva påvirker suksess?

Et utvalg av resultater ...

Størrelse?

Bør vi unngå de veldig store
IT-prosjektene?

Større prosjekter gir noe mer risiko for å feile

(%-er viser andel prosjekter opplevd som suksessfull mhp angitt faktor)

	< 10 mill	10-100 mill	> 100 mill
Nytte	31%	47%	35%
Kvalitet	24%	28%	25%
Budsjett	24%	47%	47%
Tid	29%	35%	35%
Effektivitet	24%	12%	24%

- Ingen klar sammenheng mellom budsjett-størrelse og andel prosjekter som er suksessfulle *av de som fullfører*.
- **MEN**, de store (> 100 mill) var sterkt overrepresentert i gruppen av fiaskoprosjekter, dvs prosjekter som aldri ble fullført eller som leverer så å si ingen nytte! (2-3 ganger hyppighet).
- **Stor outsourcing-undersøkelsen**: Tidobling av størrelsen ser ut doblet risikoen for å mislykkes helt.
- **NB**: Hva som er stort og lite er i realiteten **relativt** til størrelsen på man har gjort før og har mye erfaring med.

Flere resultater om størrelse

- De største prosjektene har typisk størst hyppighet av **store** problemer (23%) – og feiler altså helt mye oftere - men er også de med størst hyppighet av bra utfall (77%). Litt enten eller, ser det ut som.
- De minste prosjekter har oftere lav prioritet og lavere bestiller-kompetanse.
- De største prosjektene har oftere høy prioritet, men på den annen side oftere stor organisasjonsmessig kompleksitet hos kunde, leverandør og mottager/bruker.
- **Oppsummert**:
 - **"Tenk stort, start smått"**
 - og/eller forsøk å gjøre store prosjekter organisasjonsmessig og leveransmessig mindre komplekse – f eks ved mer smidighet og mer autonome team
 - **Trend** (som er bra): Jobb produktfokusert med kontinuerlig utvikling

Hvor mye betyr kontraktstype?

Undersøkelse

(andel prosjekter opplevd som suksessfull mhp angitt faktor)

	Fastpris	Per time	"Smidig"	Risikodeling
Nytte	0%	59%	29%	22%
Kvalitet	22%	24%	43%	22%
Budsjett	33%	31%	71%	22%
Tid	11%	29%	43%	44%
Effektivitet	0%	19%	29%	33%
Andel	18%	37%	14%	41%

Målpris med og uten øvre tak

Table 1: Success of projects using target price contracts with and without upper limit

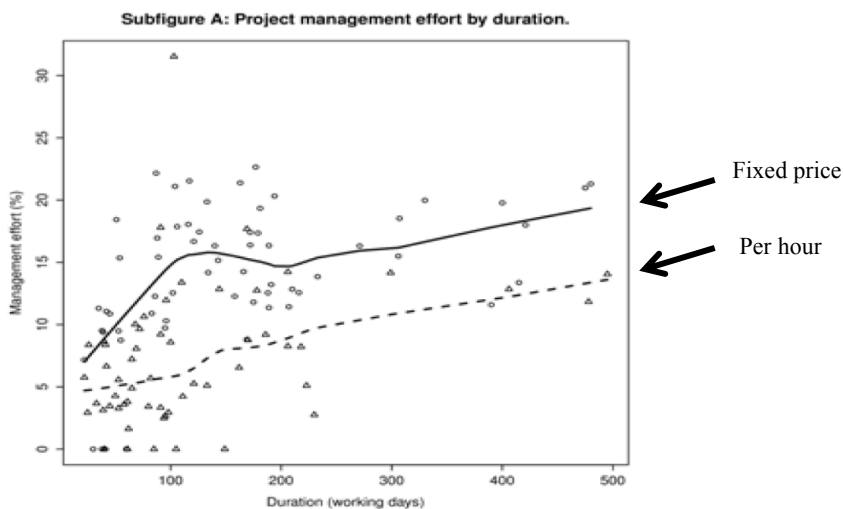
Contract type	Success criteria	Successful	Acceptable	Problematic
Target price with upper limit	Client benefits	33%	48%	19%
	Cost control	33%	33%	33%
	Time control	33%	19%	48%
Target price without upper limit	Client benefits	83%	11%	6%
	Cost control	61%	22%	17%
	Time control	63%	16%	21%

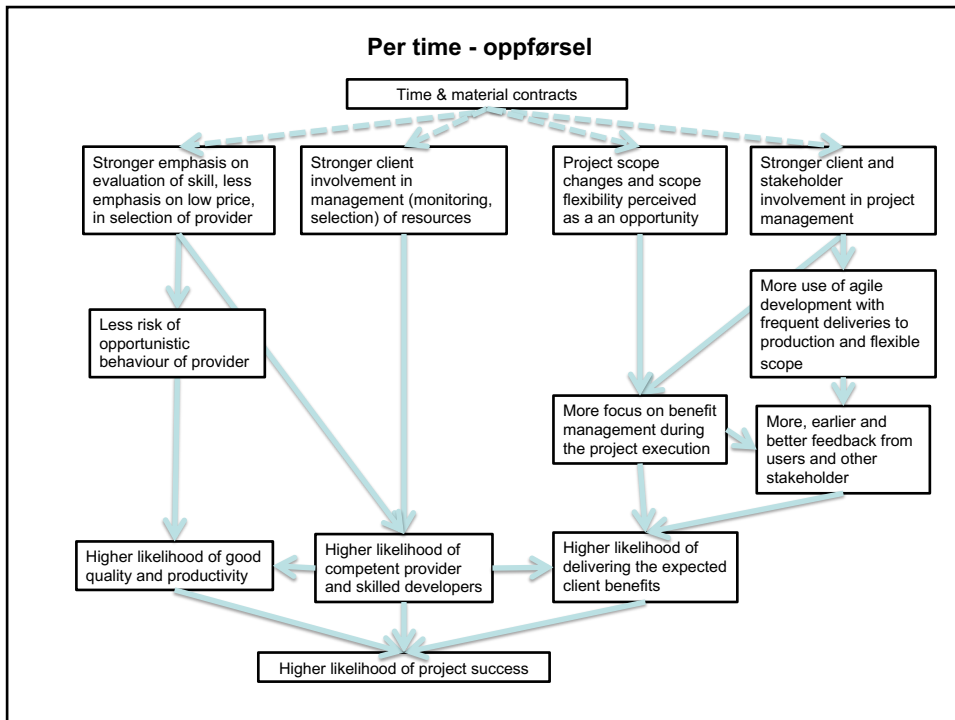
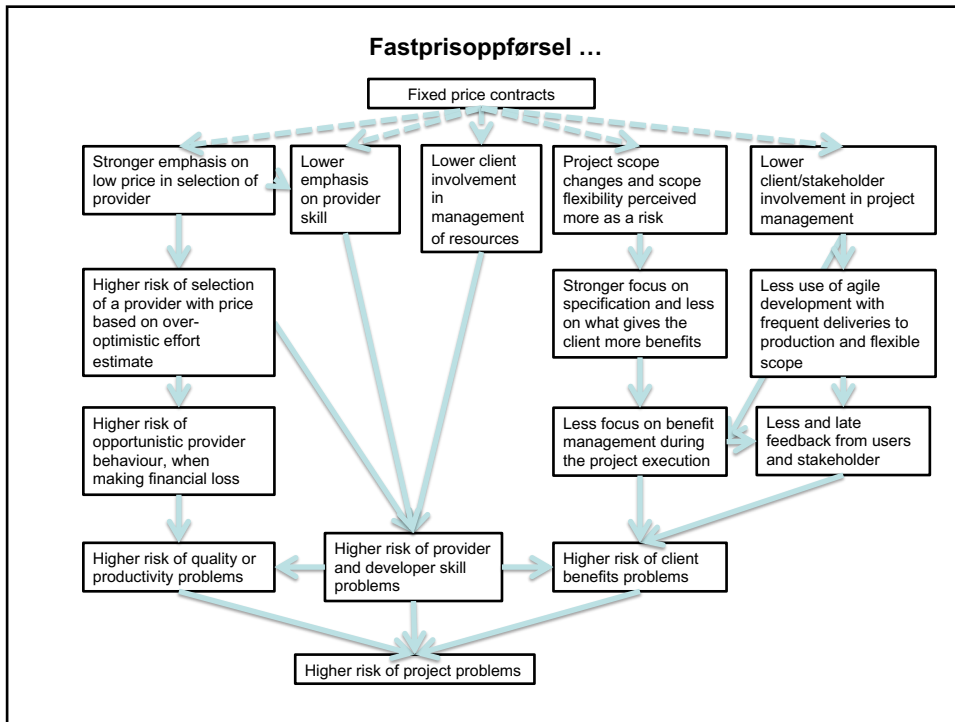
... 90% var enige i at målpris med øvre tak medførte mer fokus på hva som sto i kravspesifikasjon og mindre på hva som ga mest kundenytte.

... 68% enige i at målpris med øvre tak ga mer arbeid (som kunde) til administrasjon og oppfølging

Fastpris gir mer administrasjon

Ahonen, Jarmo J., et al. "Reported project management effort, project size, and contract type." *Journal of Systems and Software* 109 (2015): 205-213.



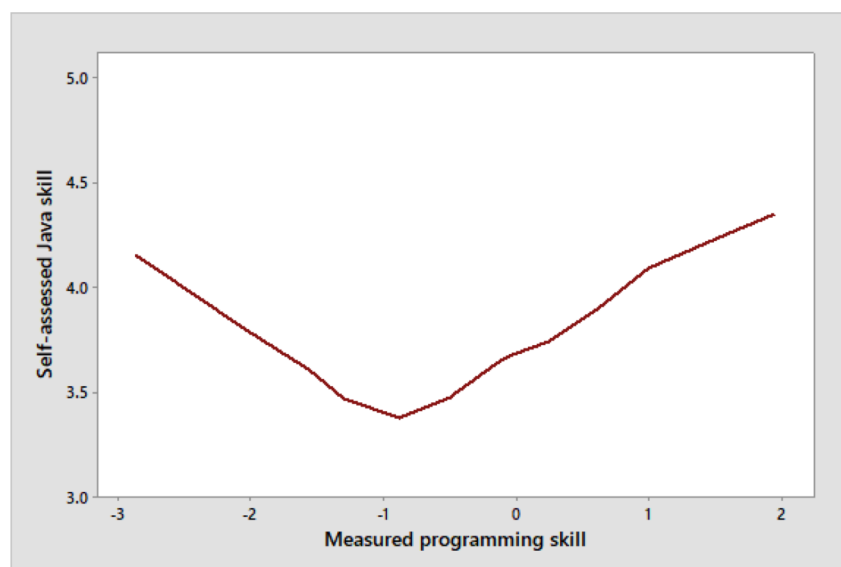
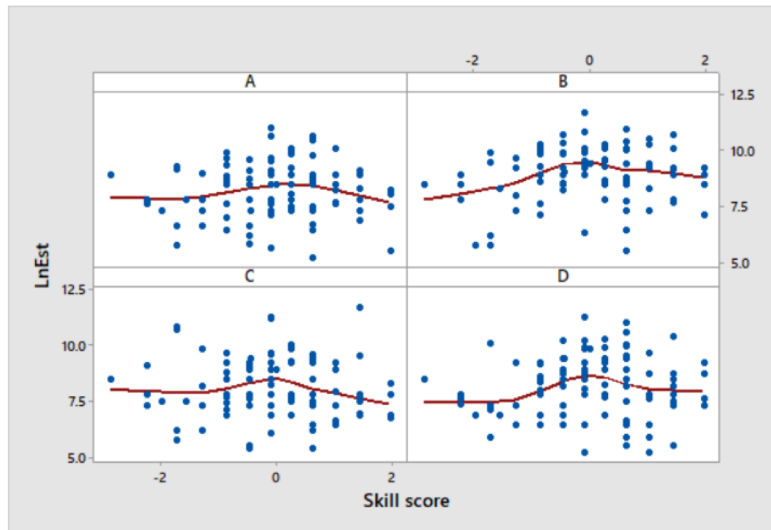


Styre ressursene selv (som kunde) eller la leverandør styre utviklingsressursene?

Undersøkelse: Ressursstyring

- Prosjektene som styrte bemanningen på personnivå var de desidert mest vellykkede (kun 8% fikk problemer). Mer krevende for kunde, men opplevdes å være lønnsomt.
- Fastpris og målpris opplevdes ofte som begrensende for kundestyring av ressursbruk ("*fastpris og målpris betyr at leverandør må få bestemme hvem som skal gjøre jobben*")
- Prosjekter med en stor leverandør og fastpris-aktig kontrakt fungerte dårligst (67% fikk problemer)
- De mest vellykkede prosjektene hadde (oppsummert) typisk:
 - Kompetente og involverte kunder
 - Bruk av per time-baserte kontrakter
 - Valgte av ressurser fra flere leverandører (og helst gode tester av kompetanse, eller bruk av uavhengige, svært relevante referanser)

Helt fersk forskning: Vurdering av kompetanse Dunning-Kruger effekt



Hjelper det å jobbe mer smidig?

Og hvordan gå fra usmidig til mer smidig?

HIT-undersøkelsen

Økning (prosentpoeng) i andel vellykkede prosjekter

	«Smidig»	Hyppig leveranse til produksjon	Fleksibelt innhold
Nytte	16%	22%	29%
Funksjonalitet	22%	29%	16%
Kvalitet	21%	6%	32%
Budsjettkontroll	2%	22%	29%
Tidskontroll	8%	11%	24%
Effektivitet	11%	5%	24%

Prosjekter som jobbet smidig, men ikke hadde hyppige leveranser til produksjon (eller grundig evaluering av kunde var MINDRE vellykket enn ikke-smidige prosjekter! Smidig er ikke smidig ...

Smidige metoder passer også godt for store prosjekter (om noen skulle tro noe annet)

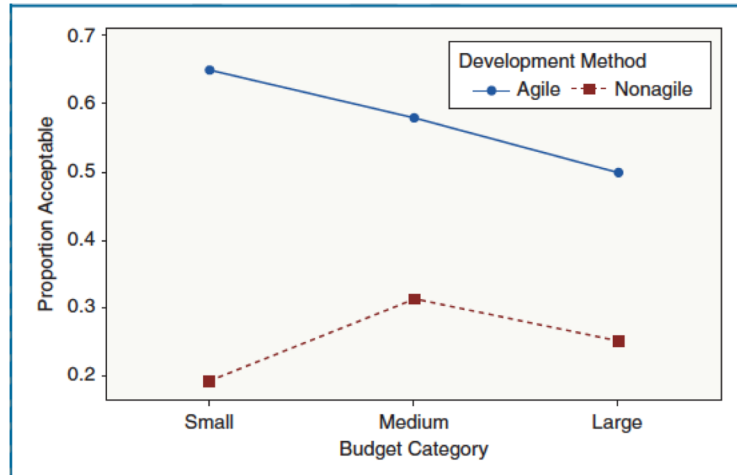


Table 3. The proportion of projects with acceptable performance.*

Factor	Category	Agile (n = 146)	Nonagile (n = 50)
Requirement volatility	High (n = 80)	58%	13%
	Low (n = 97)	61%	29%
Delivery frequency	>4 per year (n = 99)	70%	25%
	≤4 per year (n = 60)	49%	21%
Scope flexibility	High (n = 71)	85%	33%
	Low (n = 26)	50%	40%
Detail of project plan	High (n = 59)	67%	18%
	Low (n = 81)	53%	21%
Detail of requirement specification	High (n = 63)	55%	13%
	Low (n = 76)	61%	26%

Fra usmidig til mer smidig

- De som lykkes godt med smidige leveranser har tatt reisen stegvis, og jobber kontinuerlig med å bli mer smidige.
- Reisene har vært like mye kulturendringsreiser, som prosessendringsreiser. Elementer som må tas over tid inkluderer:
 - Økt fleksibilitet og læring i leveranser
 - Tidligere og hyppigere leveranser (som ofte krever nye prosesser for testing)
 - Underveis nyttestyring rundt leveranser (som bla krever kompetanse som produkteier)
 - Fra prosjekt til produktfokus (som krever mye av forretningsiden)
 - Mer autonome team (som kan kreve store omstillinger i applikasjonsarkitektur, og tillitsforhold)
 - En smidig finansiering av kontinuerlig utvikling (“fullfinansiering” og “lever nytte”, i stedet for budsjetter og kravspesifikasjoner)

Hjelper det med nyttestyring?

(og hva er nå nyttestyring?)

Undersøkelser: Nytestyring

HIT-undersøkelse:

Nytestyring-praksiser	Andel	Økning i suksess-rate mht. levert nytte
Kost-nytteanalyser i forkant	47%	6%
Ansvarlig for nytteoppnåelse	57%	22%
Plan for nytestyring	33%	31%
Nytestyring underveis	53%	34%
Evaluering av nytte underveis, og i etterkant	31%	19%

SMIOS-undersøkelse:

	Ja	Nei / Vet ikke
Business case ved oppstart	8 av 26 (31%) med problemer	2 av 9 (22%) med problemer
Ansvarlig for gevinstrealisering	5 av 18 (28%) med problemer	5 av 17 (29%) med problemer
Plan for nytestyring	5 av 17 (29%) med problemer	5 av 18 (28%) med problemer
Nytestyring underveis	3 av 15 (20%) med problemer	7 av 20 (35%) med problemer

Viktige kjennetegn ved det suksessfulle prosjektet (oppsummert)

- God kontroll på ambisjonsnivå. Unngår "for mye" på en gang og sier nei til tillegg som vil øke risiko.
- Bruker kontrakter som unngår "fastpris-oppførsel" for leverandør (og kunde): Per time eller målpris uten øvre tak.
- Har en kunde (bestiller-team) med kompetanse til å velge og styre kompetente enkeltressurser (og har mindre fokus på lav timepris).
- Har fleksibilitet i omfang/leveranser (ikke bare må-funksjonalitet)
- Har en kunde som (minimum) er sterkt involvert i planlegging av nytte og nytestyring underveis
- God bruk av smidig utvikling (kommet langt på dette)
- Kommuniserer tidlig og godt formål og effekter («den gode historien») og tenker tidlig på (og involverer) overlevering til linje, brukere og drift

Kjennetegn ved det problematiske prosjektet

- Har for høyt ambisjonsnivå. (Bruker muligheten et stort prosjekt gir til å gjøre mer.)
- Bruker kontrakter med høy risiko for "fastpris-oppførsel" for leverandør (og kunde): Fastpris og målpris med øvre tak
- Anbudsrunder med fokus på lav pris, som leder til valg av leverandør med under-prising (eller lavere kompetanse på enkeltressurser)
- Har kunde uten god kompetanse til å velge kompetente leverandører og enkeltressurse: Lar leverandører styre ressursene uten innblanding
- Ingen eller liten fleksibilitet i omfang og ser på endringer ("scope creep") som trussel
- Liten eller ingen fokus på nyttestyring, utenom å lage "business case" ved oppstart
- Lite smidig utvikling.
- Overlevering til linje, brukere og drift tenkes på kun i siste fase, eller når prosjektet leverer.