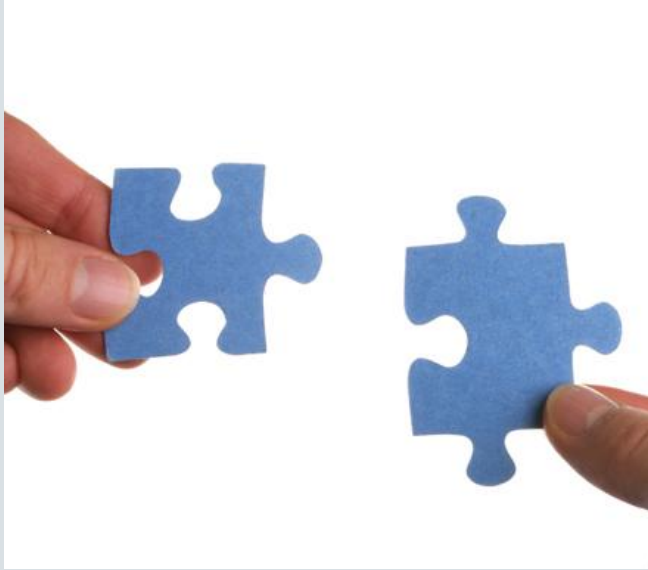


Smidig 2.0 – the need for change



Contradiction in terms - how does Agile cope with versioning?

Totto

Objectware AS

Smidig har gitt effektiv utvikling, men henger resten av verden med på det økte tempoet?

NEED FOR SPEED

Kort om dagens situasjon

- *Smidige prosjekter er et stor suksess*
- *Men vi har noen 'nye' utfordringer*
 - En lei tendens til å lage nye 2.5 lags database-sentriske siloapplikasjoner
 - "arkitektur, design er ikke viktig" – les: for vanskelig/tidkrevende
 - Testing – raske tester, som også skal være aktiv del av dokumentasjon er selvmotsigelser
 - Konfigurasjonsstyring – blir ofte 'glemt' i smidige prosjekter, siden drift sjelden er aktiv stakeholder.
 - Smidig-bevegelsen er religiøst selvsentrisk, og lite villige til å se konsekvenser
 - (Overvekt av hotshot-grooupies)

Utfordringene rundt resultatene fra Smidige prosjekter som belyst i dagens situasjon er økende, og hvis vi ikke gjør noen nødvendige justeringer, så står vi i fare for å miste den verdien vi alle har av smidig systemutvikling...

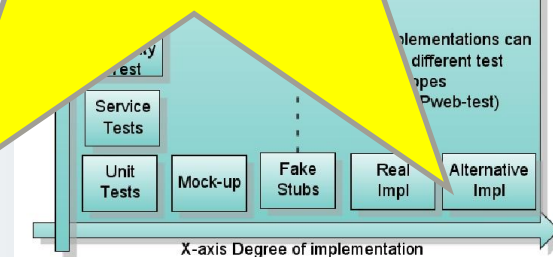
WHAT WE NEED

Hvordan håndtere utfordringene?

- Gjeninnføre arkitektur og design
 - *Tøffe utfordringer trenger de beste utviklerne!*
 - *Hvor ble det av anti-corruption layer?*
 - *Vi kan ikke fortsette å ignorere at modning tar tid*
- Opprette/standarisere test-kategorisering

Acceptance-tests				
System-tests				
Data-tests				
Service-tests				
Unit-tests				
	Mock-up	Fake	Impl A	Implementation B

Lightning
talk – ikke
tid til
detaljer...



- Gjeninnføre et bevist forhold til konfigurasjonsstyring og versjonering.
- Gjeninnføre **sunnt fornuft**

Med så store endringer på dagens praksis, ikke minst kulturelt, så ser jeg ingen annen vei enn å 'forke' bevegelsen til Smidig 2.0 for å etablere den nye retningen...

**INGEN SA NOENSINNE AT
SMIDIG VAR FOR IDIOTER**