



Jonas Andersson

@tradjonas

@umejug



1999

2007

2009

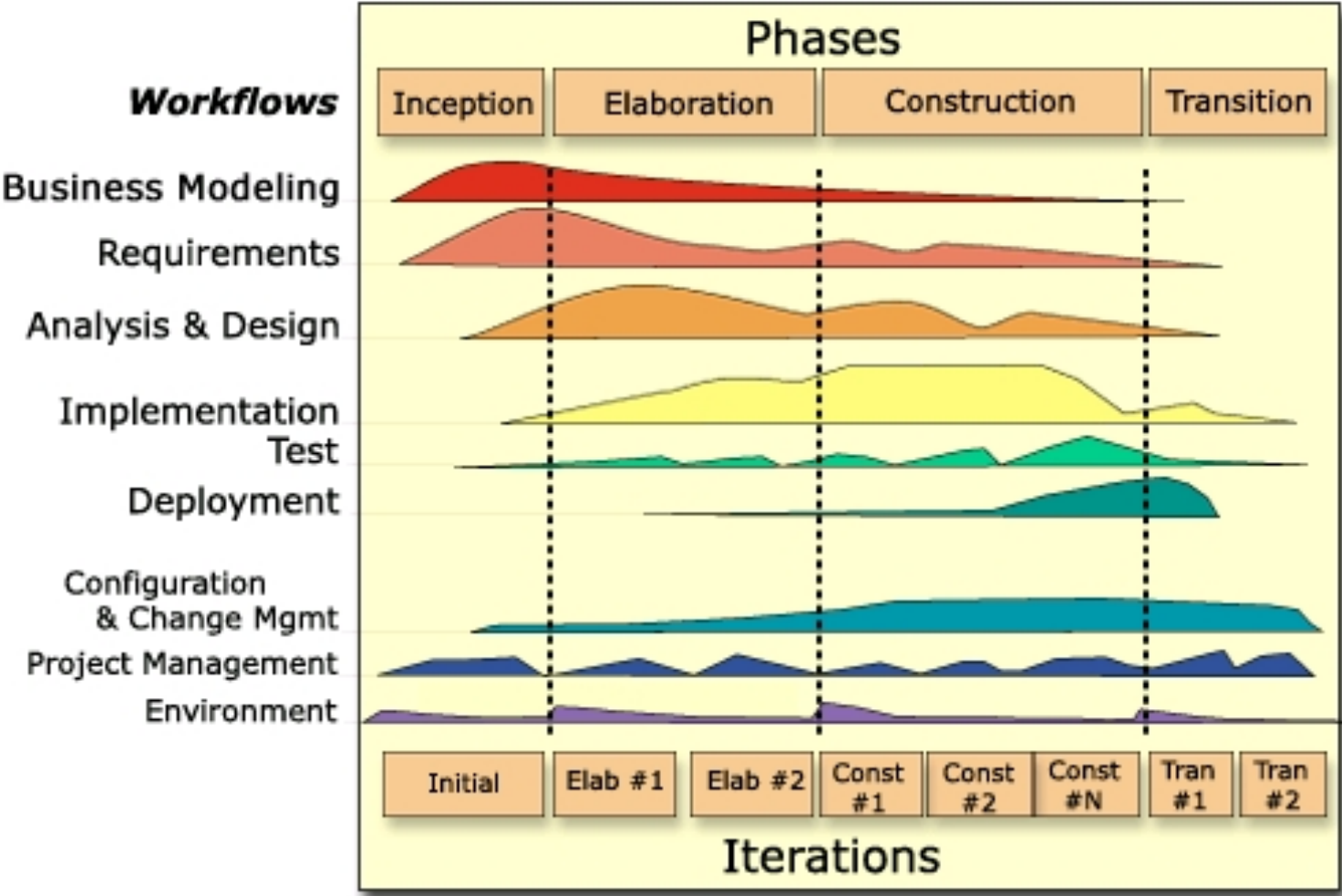


Problem i utvecklingsprojekt

- Missade tids- och kostnadsramar
- Ständig tidsnöd
- Långa oförutsägbara releasescheman
- Kvalitetssäkring fungerar dåligt
- Svårt tidsuppskatta
- Känsla av tröghet och ineffektivitet
- Oengagerad personal → inte roligt på jobbet
- osv

Utvecklingsmodell?

RUP – Rational Unified Process



Rational Unified Process

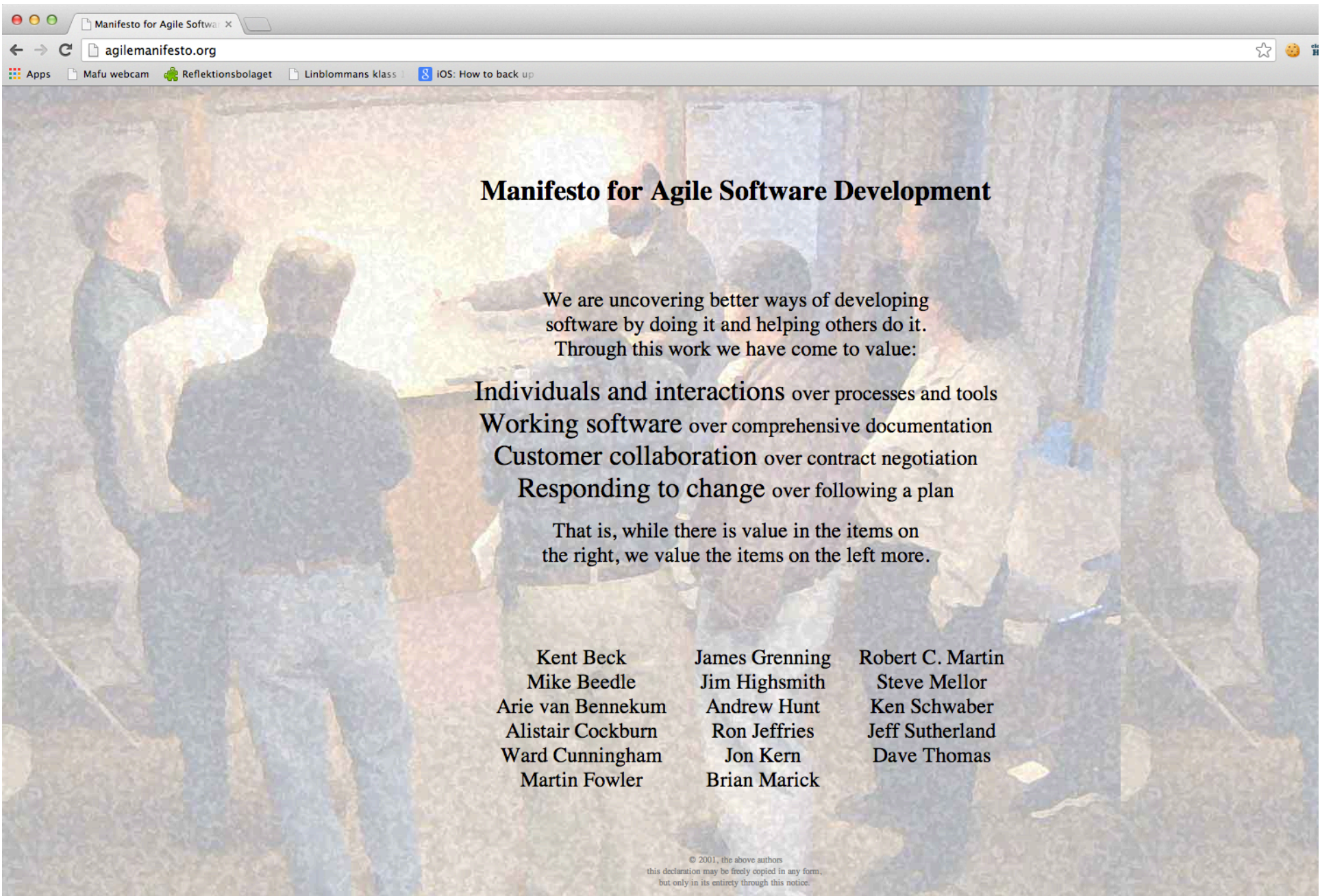
- Inception
 - Vet vi vad vi ska bygga?
- Elaboration
 - Vet vi hur vi ska bygga?
- Construction
 - Har vi byggt det?
- Transition
 - Kan användarna använda det vi byggt?

Rational Unified Process

- Mastodonten RUP
 - 4 faser
 - 9 discipliner
 - 100+ artefakter
 - ~30 roller
- Men RUP är ett **ramverk** som ska instansieras till en process som passar aktuellt projekt!

RUP:s ”Nyckelprinciper för affärsdriven utveckling”

1. Anpassa processen
2. Balansera intressenters prioriteringar
3. Samarbeta mellan team
4. Demonstrera kundvärde iterativt
5. Höj abstraktionsnivån
6. Fokusera kontinuerligt på kvalitet



Manifesto for Agile Software Development

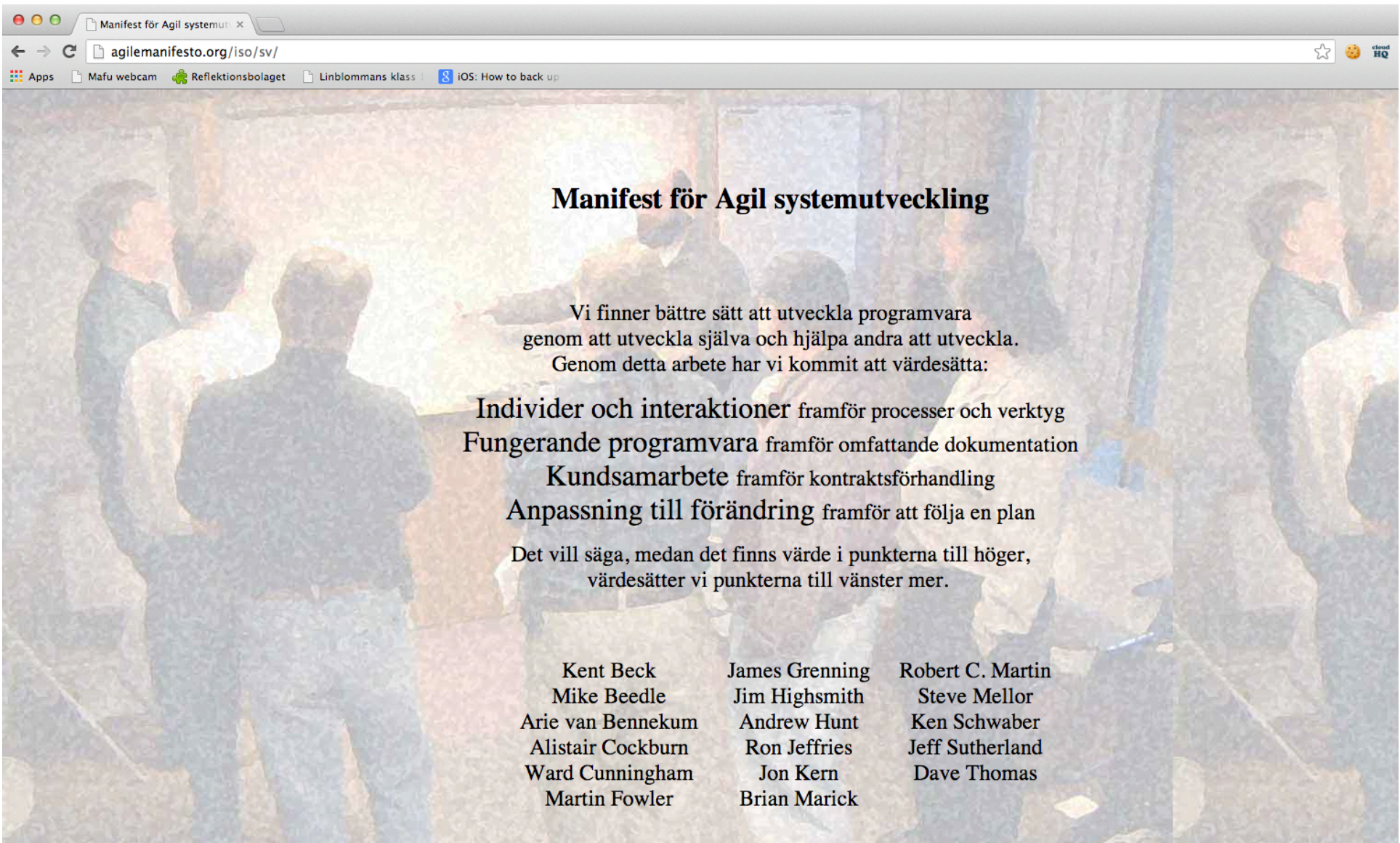
We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

© 2001, the above authors
this declaration may be freely copied in any form,
but only in its entirety through this notice.



Manifest för Agil systemutveckling

Vi finner bättre sätt att utveckla programvara genom att utveckla själva och hjälpa andra att utveckla. Genom detta arbete har vi kommit att värdesätta:

Individer och interaktioner framför processer och verktyg
Fungerande programvara framför omfattande dokumentation

Kundsamarbete framför kontraktsförhandling
Anpassning till förändring framför att följa en plan

Det vill säga, medan det finns värde i punkterna till höger, värdesätter vi punkterna till vänster mer.

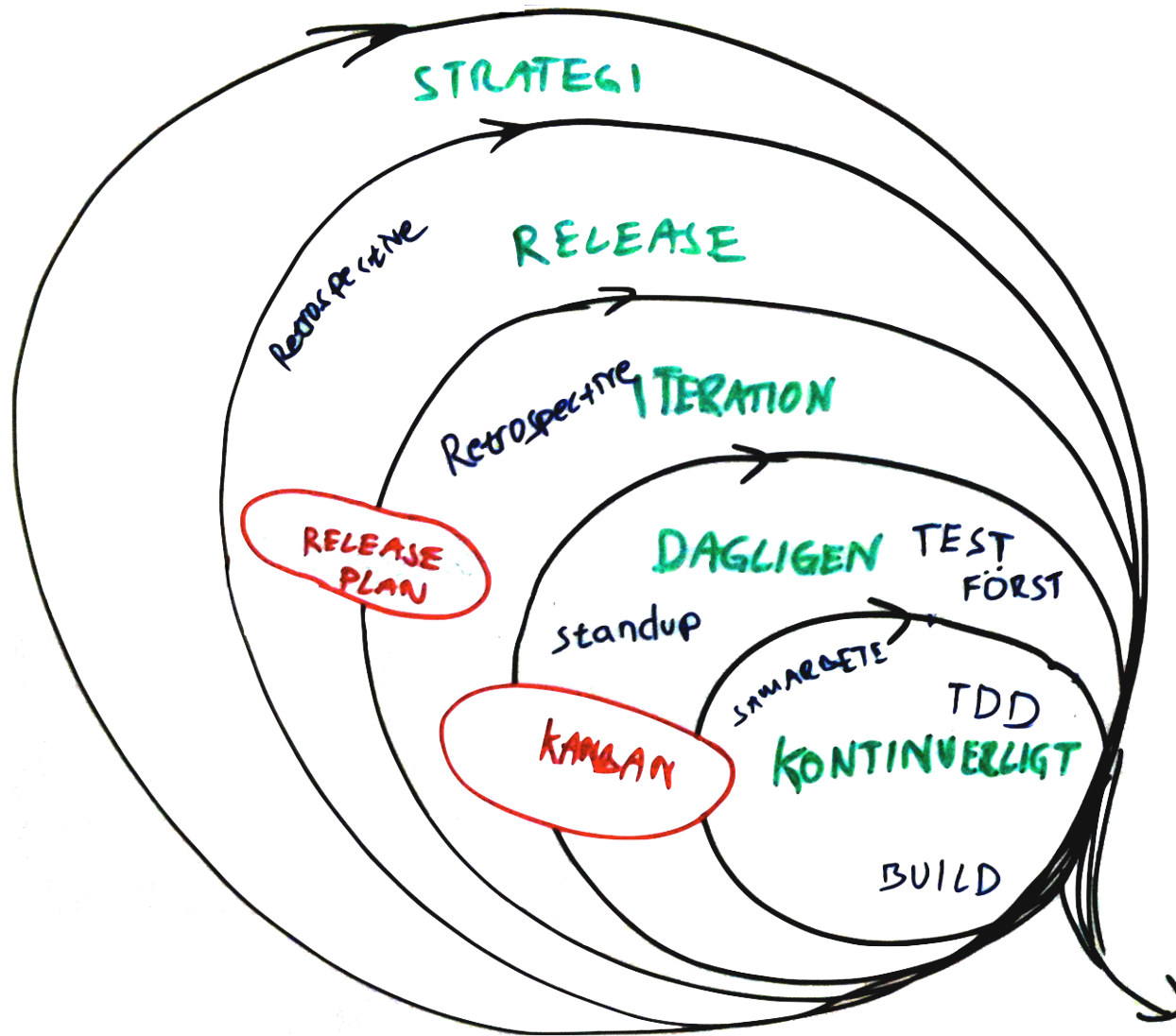
Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

Agile

Adjective

S: (adj) **agile**, nimble, quick, spry (moving quickly and lightly) "*sleek and agile as a gymnast*"; "*as nimble as a deer*"; "*nimble fingers*"; "*quick of foot*"; "*the old dog was so spry it was halfway up the stairs before we could stop it*"

S: (adj) **agile**, nimble (mentally quick) "*an agile mind*"; "*nimble wits*"



FUNGERANDE
MJKVÄRDA
=
NÖJDA KUNDER

Arbetsätt

Kanban

Continuous Integration (CI)

Test Driven Development (TDD)

Acceptance TDD (ATDD) / Behaviour Driven Development (BDD)

Automatiserade testfall med FitNesse

Parprogrammering

Dagliga möten

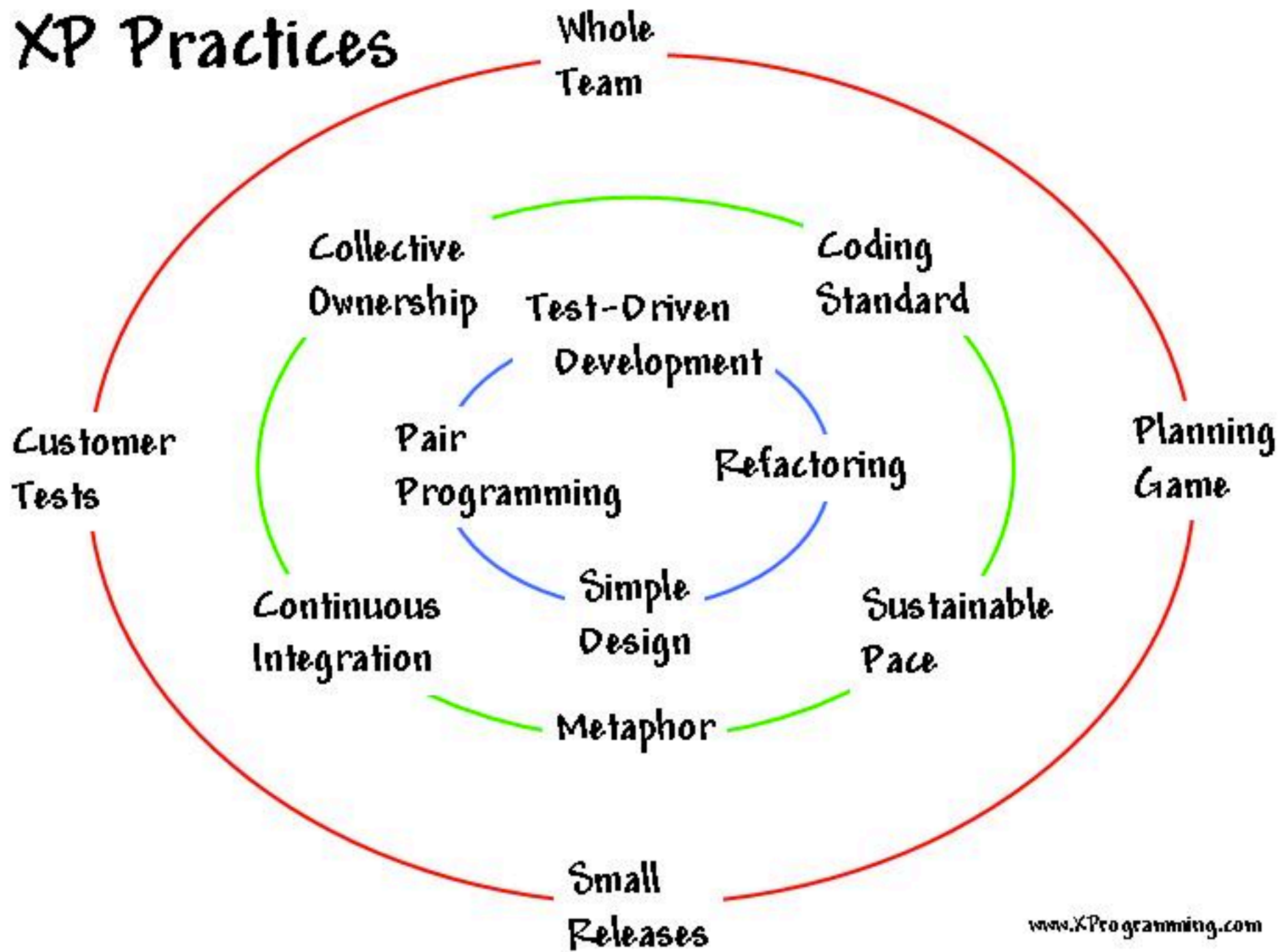
Retrospective med demo

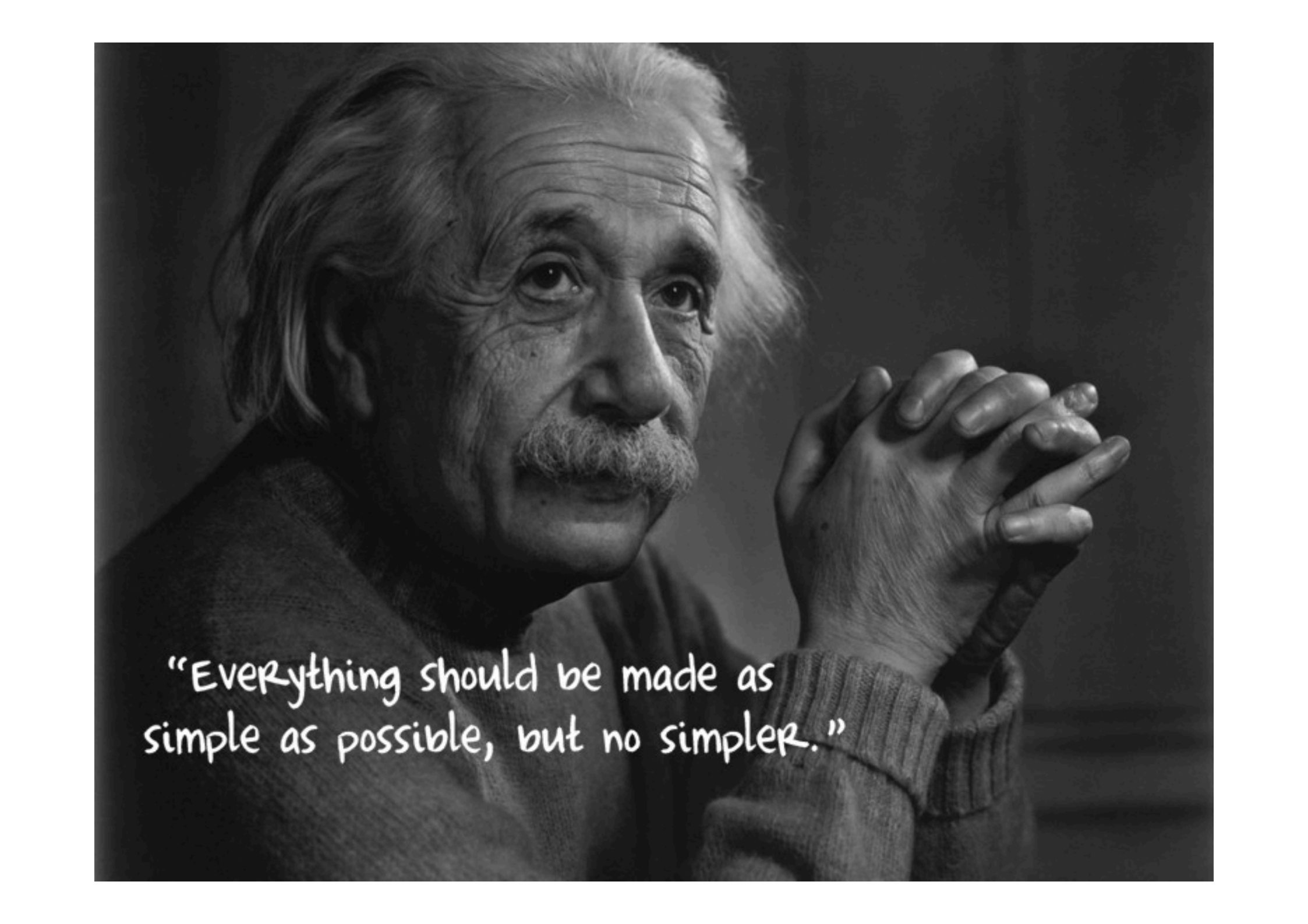
Estimering med story points

Extreme Programming (XP)

- Principer (Values)
 - Feedback
 - Communication
 - Simplicity
 - Courage

XP Practices



A black and white portrait of Albert Einstein. He is shown from the chest up, looking slightly to the right of the camera. His hair is wild and white, and he has a prominent white mustache. His hands are clasped together in front of him, resting on his lap. He is wearing a dark, textured sweater. The background is dark and out of focus.

“Everything should be made as simple as possible, but no simpler.”

Lean Software Development - principer

Minimera slöseri

Bygg in kvalitet

Skapa kunskap

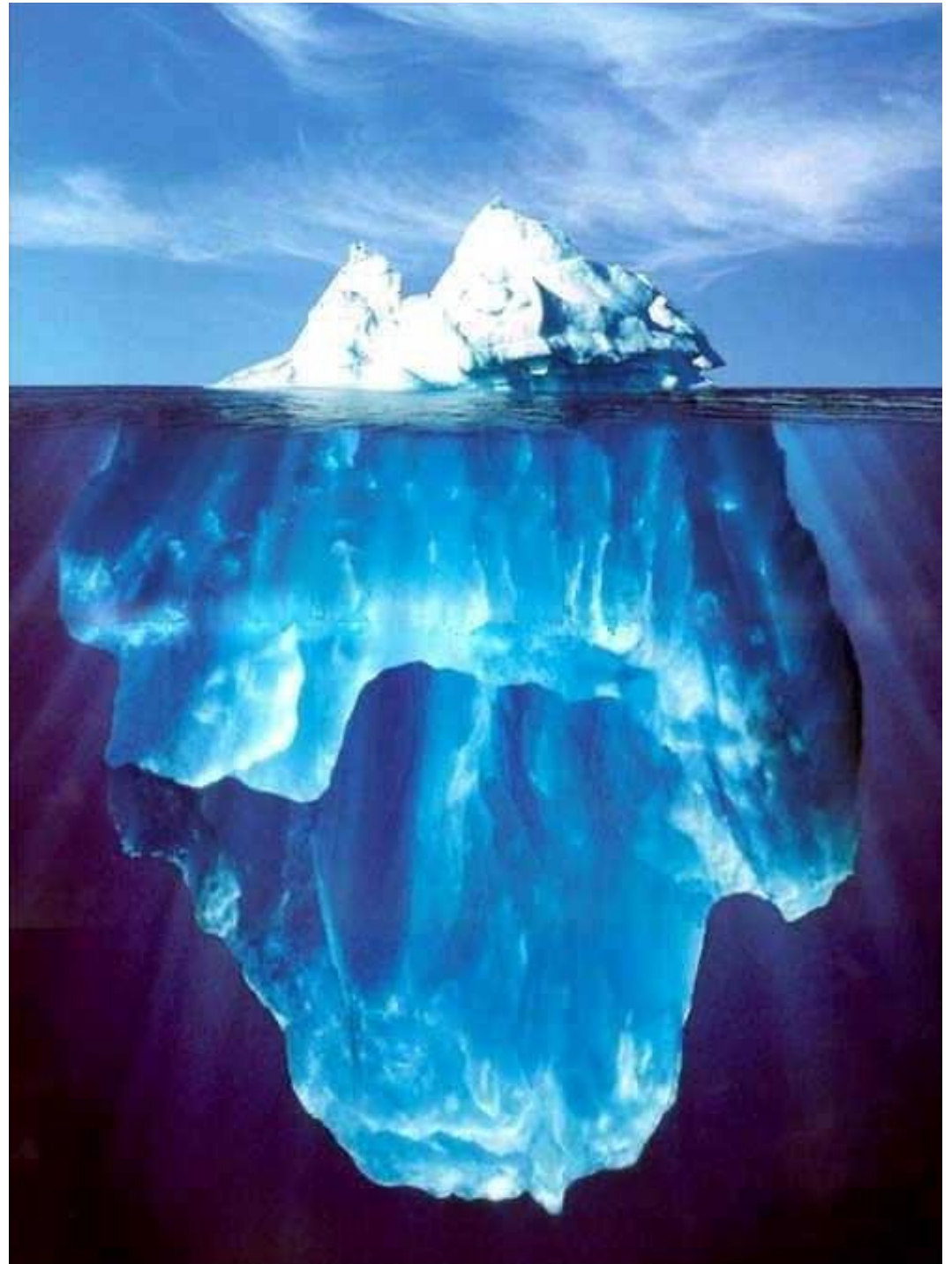
Besluta sent

Leverera snabbt

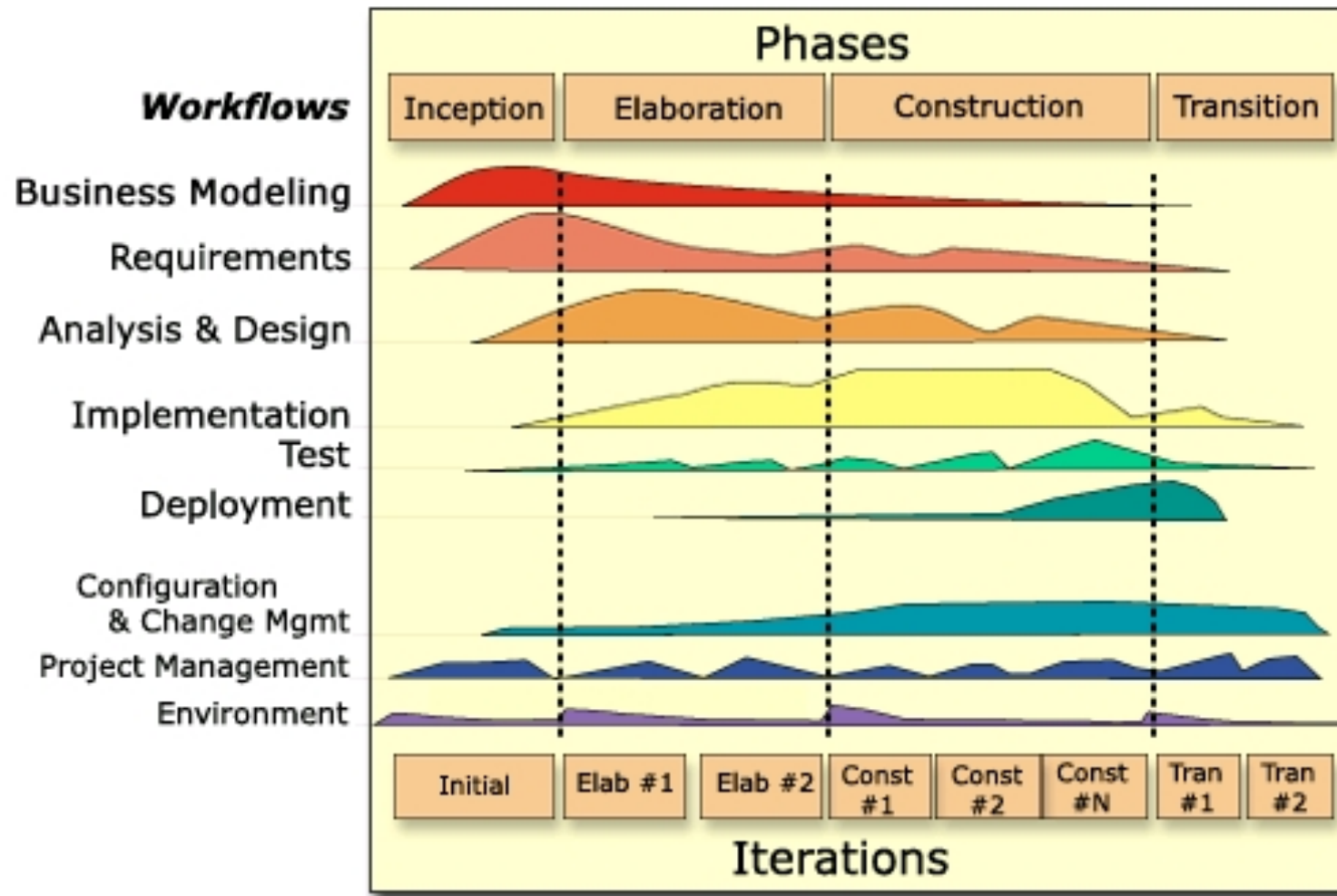
Respektera människor

Optimera helheten

Minimera slöseri



RUP vs Agila utvecklingsmodeller?



Sammanfattning

- En bra utvecklingsmodell hjälper dig/teamet/organisationen att göra rätt sak på rätt sätt.
- Det finns ett flertal att välja mellan.
- Keep it simple! Börja med ett minimum, lägg sedan till vid behov.
- Feedback!
- Agila manifestet och LSD principer hjälper dig att välja rätt arbetssätt.

Till nästa föreläsning

- Agile Product Ownership in a nutshell – 16 min.
- <http://blog.crisp.se/2012/10/25/henrikkniberg/agile-product-ownership-in-a-nutshell>
- Se den!