

# Ohjelmistotuotanto

Matti Luukkainen ja ohjaajat Valtteri Kantanen, Hannah  
Leinson, Riku Rauhala, Ville Saastamoinen

syksy 2023

Luento 10

28.11.2023

# Kurssipalaute

- ▶ Kurssipalaute
  - ▶ Kurssilla lopussa kerättävän palautteen lisäksi ns. jatkuva palaute <https://norppa.helsinki.fi>

## Kurssin kaksi viimeistä viikkoa

- ▶ Miniprojektit jatkuvat
  - ▶ tällä ja ensiviikolla asiakastapaaminen

## Kurssin kaksi viimeistä viikkoa

- ▶ Miniprojektit jatkuvat
  - ▶ tällä ja ensiviikolla asiakastapaaminen
- ▶ Loppudemot (jokainen ryhmä osallistuu toiseen demoista)
  - ▶ ti 12.12. klo 14-16 B123
  - ▶ to 14.12. klo 12-14 B123
  - ▶ ilmoittautuminen tulossa pian...

# Kurssin kaksi viimeistä viikkoa

- ▶ Miniprojektit jatkuvat
  - ▶ tällä ja ensiviikolla asiakastapaaminen
- ▶ Loppudemot (jokainen ryhmä osallistuu toiseen demoista)
  - ▶ ti 12.12. klo 14-16 B123
  - ▶ to 14.12. klo 12-14 B123
  - ▶ ilmoittautuminen tulossa pian...
- ▶ Vierailuluennot
  - ▶ ma 4.12. 12 Jami Kousa Unity, Mikko Tiainen Hogrefe
  - ▶ ti 5.12. 12 Anniina Sallinen Oura
  - ▶ ma 11.12. 12 Hannu Kokko Elisa
  - ▶ ti 12.12. 12 Kasper Hirvisalo Unity, Ville Nordberg Trail openers

# Kurssin kaksi viimeistä viikkoa

- ▶ Miniprojektit jatkuvat
  - ▶ tällä ja ensiviikolla asiakastapaaminen
- ▶ Loppudemot (jokainen ryhmä osallistuu toiseen demoista)
  - ▶ ti 12.12. klo 14-16 B123
  - ▶ to 14.12. klo 12-14 B123
  - ▶ ilmoittautuminen tulossa pian...
- ▶ Vierailuluennot
  - ▶ ma 4.12. 12 Jami Kousa Unity, Mikko Tiainen Hogrefe
  - ▶ ti 5.12. 12 Anniina Sallinen Oura
  - ▶ ma 11.12. 12 Hannu Kokko Elisa
  - ▶ ti 12.12. 12 Kasper Hirvisalo Unity, Ville Nordberg Trail openers
- ▶ Koe tiistaina 19.12. klo 7-14 moodlessa (koeaika 4h)

Laajan mittakaavan ketterä

## Laajan mittakaavan ketterä

- ▶ Ketterä tarkoitettu alunperin pienien tiimien hallintaan
  - ▶ Scrum: 3-9 kehittäjää
  - ▶ Entä jos on kyseessä massiivinen softa?

## Laajan mittakaavan ketterä

- ▶ Ketterä tarkoitettu alunperin pienien tiimien hallintaan
  - ▶ Scrum: 3-9 kehittäjää
  - ▶ Entä jos on kyseessä massiivinen softa?
- ▶ Perusperiaate pitää **tiimit pieninä**, ja kasvattaa tuotantokapasiteettia käyttämällä **useampia tiimejä**
  - ▶ Tämä edellyttää, tiimien välistä töiden koordinointia

## Laajan mittakaavan ketterä

- ▶ Ketterä tarkoitettu alunperin pienien tiimien hallintaan
  - ▶ Scrum: 3-9 kehittäjää
  - ▶ Entä jos on kyseessä massiivinen softa?
- ▶ Perusperiaate pitää **tiimit pieninä**, ja kasvattaa tuotantokapasiteettia käyttämällä **useampia tiimejä**
  - ▶ Tämä edellyttää, tiimien välistä töiden koordinointia
- ▶ Jo kauan käytetty tapa: *Scrum of Scrums*
  - ▶ koordinoiva tiimi, johon kuuluu jäseniä jokaisesta Scrum-tiimistä
  - ▶ esim. scrum master tai lead developer osallistuu

## Laajan mittakaavan ketterä

- ▶ Ketterä tarkoitettu alunperin pienien tiimien hallintaan
  - ▶ Scrum: 3-9 kehittäjää
  - ▶ Entä jos on kyseessä massiivinen softa?
- ▶ Perusperiaate pitää **tiimit pieninä**, ja kasvattaa tuotantokapasiteettia käyttämällä **useampia tiimejä**
  - ▶ Tämä edellyttää, tiimien välistä töiden koordinointia
- ▶ Jo kauan käytetty tapa: *Scrum of Scrums*
  - ▶ koordinoiva tiimi, johon kuuluu jäseniä jokaisesta Scrum-tiimistä
  - ▶ esim. scrum master tai lead developer osallistuu
- ▶ Scrum of Scrums -tiimi voi tavata päivittäin/viikoittain

## Scrum of Scrums

- ▶ Scrum of Scrums -periaate on jo hyvin vanha
  - ▶ Jeff Sutherland 1996

## Scrum of Scrums

- ▶ Scrum of Scrums -periaate on jo hyvin vanha
  - ▶ Jeff Sutherland 1996
- ▶ Fimassa satoja sovelluskehittäjiä, kymmeniä Scrum-tiimejä ja useita eri tuotteita
  - ▶ Jokaisen tuotteen tiimejä kordinoi kerran viikossa kokoontuva Scrum of Scrums

## Scrum of Scrums

- ▶ Scrum of Scrums -periaate on jo hyvin vanha
  - ▶ Jeff Sutherland 1996
- ▶ Fimassa satoja sovelluskehittäjiä, kymmeniä Scrum-tiimejä ja useita eri tuotteita
  - ▶ Jokaisen tuotteen tiimejä kordinoi kerran viikossa kokoontuva Scrum of Scrums
- ▶ Koko tuotejoukkoa hallinnoi kuukausittain kokoontuva "management Scrum"
  - ▶ koostui yrityksen johdosta, tuotepäälliköistä ja johtavista ohjelmistoarkkitehdeistä

# Scrum of Scrums

- ▶ Scrum of Scrums -periaate on jo hyvin vanha
  - ▶ Jeff Sutherland 1996
- ▶ Fimassa satoja sovelluskehittäjiä, kymmeniä Scrum-tiimejä ja useita eri tuotteita
  - ▶ Jokaisen tuotteen tiimejä kordinoi kerran viikossa kokoontuva Scrum of Scrums
- ▶ Koko tuotejoukkoa hallinnoi kuukausittain kokoontuva "management Scrum"
  - ▶ koostui yrityksen johdosta, tuotepäälliköistä ja johtavista ohjelmistoarkkitehdeistä
- ▶ Kuvaus ei ole kovin seikkaperäinen
  - ▶ miten esim. backlogien suhteiden tulisi toimia?

## Laajan mittakaavan ketterä

- ▶ Viimeisen kymmenen vuoden aikana ketterän skaalaamiseen esitellyt useita menetelmiä, esim.
  - ▶ Scaled Agile Framework eli SAFe
  - ▶ Large Scale Scrum eli LeSS
  - ▶ Disiplined Agile eli DA
  - ▶ Enterprise Scrum
  - ▶ Nexus

## Laajan mittakaavan ketterä

- ▶ Viimeisen kymmenen vuoden aikana ketterän skaalaamiseen esitelty useita menetelmiä, esim.
  - ▶ Scaled Agile Framework eli SAFe
  - ▶ Large Scale Scrum eli LeSS
  - ▶ Disiplined Agile eli DA
  - ▶ Enterprise Scrum
  - ▶ Nexus
- ▶ Laajentavat ketteryyttä ottamalla mukaan lean-ajattelua

## Laajan mittakaavan ketterä

- ▶ Viimeisen kymmenen vuoden aikana ketterän skaalaamiseen esitelty useita menetelmiä, esim.
  - ▶ Scaled Agile Framework eli SAFe
  - ▶ Large Scale Scrum eli LeSS
  - ▶ Disiplined Agile eli DA
  - ▶ Enterprise Scrum
  - ▶ Nexus
- ▶ Laajentavat ketteryyttä ottamalla mukaan lean-ajattelua
- ▶ Toisin kuin ketterä, lean on lähtökohtaisesti tarkoitettu toimimaan suressa skaalassa
  - ▶ sisältää enemmän koko organisaation toimintaa ohjaavia periaatteita kuin perinteinen ketterä

## Laajan mittakaavan ketterä

- ▶ Viimeisen kymmenen vuoden aikana ketterän skaalaamiseen esitelyt useita menetelmiä, esim.
  - ▶ **Scaled Agile Framework eli SAFe**
  - ▶ **Large Scale Scrum eli LeSS**
  - ▶ **Disiplined Agile eli DA**
  - ▶ **Enterprise Scrum**
  - ▶ **Nexus**
- ▶ Laajentavat ketteryyttä ottamalla mukaan lean-ajattelua
- ▶ Toisin kuin ketterä, lean on lähtökohtaisesti tarkoitettu toimimaan suuressa skaalassa
  - ▶ sisältää enemmän koko organisaation toimintaa ohjaavia periaatteita kuin perinteinen ketterä

# SAFe eli Scaled Agile Framework

# SAFe eli Scaled Agile Framework

- ▶ Pääasiallinen kehittäjä on David Leffingwell joka toimi Nokia Mobile Phonesissa (NMP) konstulttina 2000-luvulla
  - ▶ SAFe on syntynyt pitkälti Nokialla tehdyn työn pohjalta
  - ▶ NMP:lla olikin käytössä eräänlainen esiversio SAFe:sta

# SAFe eli Scaled Agile Framework

- ▶ Pääasiallinen kehittäjä on David Leffingwell joka toimi Nokia Mobile Phonesissa (NMP) konstulttina 2000-luvulla
  - ▶ SAFe on syntynyt pitkälti Nokialla tehdyn työn pohjalta
  - ▶ NMP:lla olikin käytössä eräänlainen esiversio SAFe:sta
- ▶ SAFe:n virallinen ensimmäinen version julkaistiin 2011

# SAFe eli Scaled Agile Framework

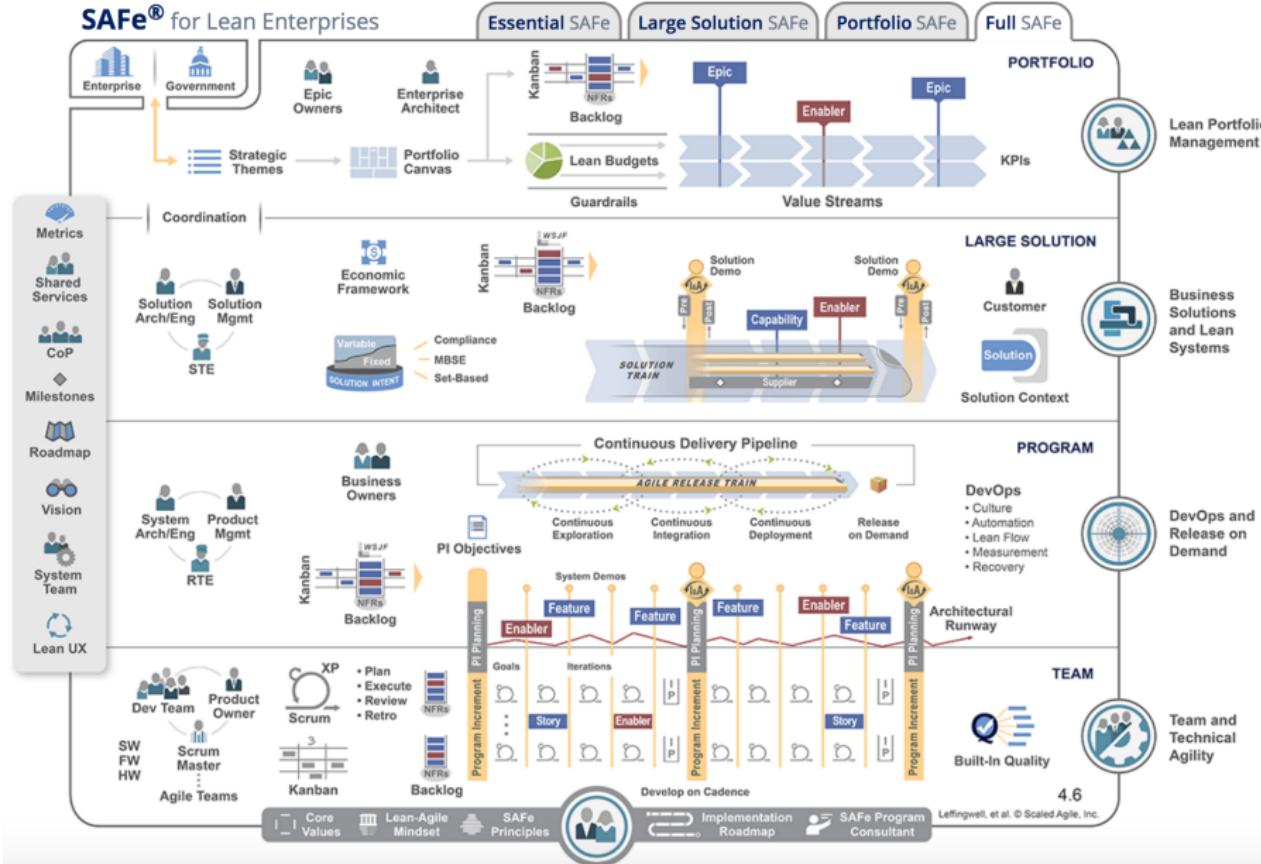
- ▶ Pääasiallinen kehittäjä on David Leffingwell joka toimi Nokia Mobile Phonesissa (NMP) konstulttina 2000-luvulla
  - ▶ SAFe on syntynyt pitkälti Nokialla tehdyn työn pohjalta
  - ▶ NMP:lla olikin käytössä eräänlainen esiversio SAFe:sta
- ▶ SAFe:n virallinen ensimmäinen version julkaistiin 2011
- ▶ Yhdistää *kaikki* viime vuosien ketterän ja leanin parhaat käytänteet sekä joukon tuotteiden hallinnointiperiaatteita

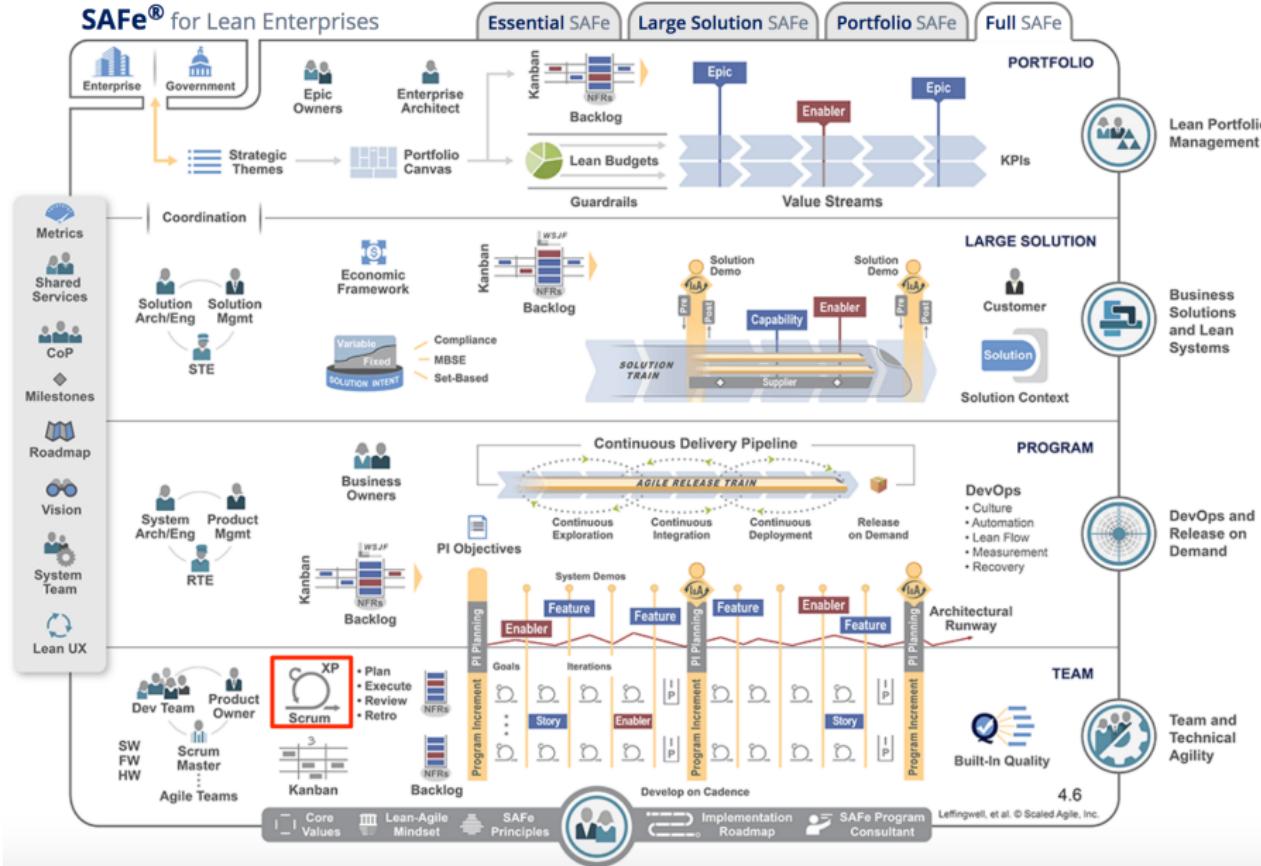
# SAFe eli Scaled Agile Framework

- ▶ Pääasiallinen kehittäjä on David Leffingwell joka toimi Nokia Mobile Phonesissa (NMP) konstulttina 2000-luvulla
  - ▶ SAFe on syntynyt pitkälti Nokialla tehdyn työn pohjalta
  - ▶ NMP:lla olikin käytössä eräänlainen esiversio SAFe:sta
- ▶ SAFe:n virallinen ensimmäinen version julkaistiin 2011
- ▶ Yhdistää *kaikki* viime vuosien ketterän ja leanin parhaat käytänteet sekä joukon tuotteiden hallinnointiperiaatteita
- ▶ SAFe tarjoaa suuren määrän käytänteitä, henkilö- ja tiimirooleja sekä käsitteitä
  - ▶ *menetelmäkehys*, yritykset räätälöivät itselleen sopivanlaisen prosessin käyttäen SAFe:n tarjoamia työkaluja

# SAFe eli Scaled Agile Framework

- ▶ Pääasiallinen kehittäjä on David Leffingwell joka toimi Nokia Mobile Phonesissa (NMP) konstulttina 2000-luvulla
  - ▶ SAFe on syntynyt pitkälti Nokialla tehdyn työn pohjalta
  - ▶ NMP:lla olikin käytössä eräänlainen esiversio SAFe:sta
- ▶ SAFe:n virallinen ensimmäinen version julkaistiin 2011
- ▶ Yhdistää *kaikki* viime vuosien ketterän ja leanin parhaat käytänteet sekä joukon tuotteiden hallinnointiperiaatteita
- ▶ SAFe tarjoaa suuren määrän käytänteitä, henkilö- ja tiimirooleja sekä käsitteitä
  - ▶ *menetelmäkehys*, yritykset räätälöivät itselleen sopivanlaisen prosessin käyttäen SAFe:n tarjoamia työkaluja
- ▶ Tarojaan 4 erikokoista valmiaksi räätälöityä konfiguraatiota
  - ▶ *Essential SAFe*: pienemmille yrityksille ja SAFen soveltamisen alkuvaiheeseen
  - ▶ *Full SAFe*: massiivisten, useita eri tuotteita hallitsevan yrityksen käyttöön



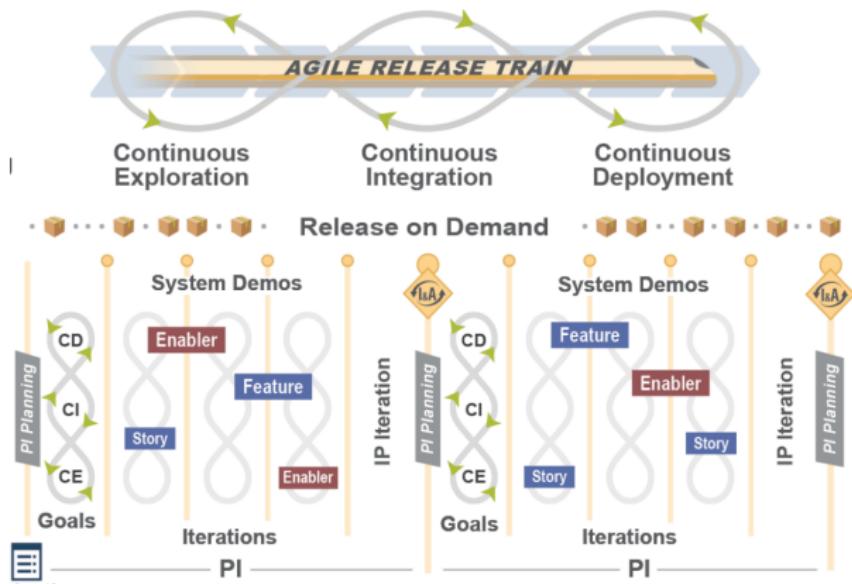


## Release train ja product increment

- ▶ Sovelluskehityksen ytimessä modifioitu Scrum

# Release train ja product increment

- ▶ Sovelluskehityksen ytimessä modifioitu Scrum
- ▶ Samaa tuotetta tekevät tiimit muodostavat *release trainin*
  - ▶ tuottavat yhdessä isompia kokonaisuuksia useammasta sprintistä koostuvan *product increment*-ajanjakson aikana



- ▶ Product incrementtejä ja release traineja taas ohjaillaan yhä korkeammalta organisaatiosta erilaisten johtajien toimesta
  - ▶ SAFe tarjoaa tähän paljon tukea käsittelijöiden ja rooliensä kautta

- ▶ Product incrementtejä ja release traineja taas ohjaillaan yhä korkeammalta organisaatiosta erilaisten johtajien toimesta
  - ▶ SAFe tarjoaa tähän paljon tukea käsittelystön ja roolien kautta
- ▶ SAFe dokumentoituu laajasti ja tarjoaa tarkat ohjeet SAFen käyttöönottoon ja noudattamiseen

- ▶ Product incrementtejä ja release traineja taas ohjaillaan yhä korkeammalta organisaatiosta erilaisten johtajien toimesta
  - ▶ SAFe tarjoaa tähän paljon tukea käsittelystön ja roolien kautta
- ▶ SAFe dokumentoituu laajasti ja tarjoaa tarkat ohjeet SAFen käyttöönnottoon ja noudattamiseen
- ▶ SAFe firmojen johdon suosiossa
  - ▶ tarjoaaakin firman managementille sopivasti tekemistä roolien ja käytänteiden muodossa

# SAFe ja kriitikki

## SAFe ja kriitikki

- ▶ SAFe sisältää käytännössä kaikki mahdolliset ketterän ja lean-ohjelmistokehityksen parhaat käytänteet
  - ▶ Kaikki vieläpä selkeästi ja yksityiskohtaisesti dokumentoituina

## SAFe ja kriitikki

- ▶ SAFe sisältää käytännössä kaikki mahdolliset ketterän ja lean-ohjelmistokehityksen parhaat käytänteet
  - ▶ Kaikki vieläpä selkeästi ja yksityiskohtaisesti dokumentoituina
- ▶ Ketterän- ja lean-kehityksen supermarket, kaikki on helposti saatavissa valmiina pakatussa mudossa
  - ▶ Pick and mix, avaa paketti ja seuraa ohjetta...

## SAFe ja kriitikki

- ▶ SAFe sisältää käytännössä kaikki mahdolliset ketterän ja lean-ohjelmistokehityksen parhaat käytänteet
  - ▶ Kaikki vieläpä selkeästi ja yksityiskohtaisesti dokumentoituina
- ▶ Ketterän- ja lean-kehityksen supermarket, kaikki on helposti saatavissa valmiina pakatussa mudossa
  - ▶ Pick and mix, avaa paketti ja seuraa ohjetta...
- ▶ SAFe käytetään paljon ja se on erityisen suosittu Suomessa

# SAFe ja kriitikki

- ▶ SAFe sisältää käytännössä kaikki mahdolliset ketterän ja lean-ohjelmistokehityksen parhaat käytänteet
  - ▶ Kaikki vieläpä selkeästi ja yksityiskohtaisesti dokumentoituina
- ▶ Ketterän- ja lean-kehityksen supermarket, kaikki on helposti saatavissa valmiina pakatussa mudossa
  - ▶ Pick and mix, avaa paketti ja seuraa ohjetta...
- ▶ SAFe käytetään paljon ja se on erityisen suosittu Suomessa
- ▶ SAFe on saanut osakseen myös paljon kritiikkiä
  - ▶ Osa kritiikistä kohdistuu SAFen määrittelemän prosessin raskauteen
  - ▶ Osa taas SAFe:n top down -management luonteeseen

# SAFe ja kriitikki

- ▶ SAFe sisältää käytännössä kaikki mahdolliset ketterän ja lean-ohjelmistokehityksen parhaat käytänteet
  - ▶ Kaikki vieläpä selkeästi ja yksityiskohtaisesti dokumentoituina
- ▶ Ketterän- ja lean-kehityksen supermarket, kaikki on helposti saatavissa valmiina pakatussa mudossa
  - ▶ Pick and mix, avaa paketti ja seuraa ohjetta...
- ▶ SAFe käytetään paljon ja se on erityisen suosittu Suomessa
- ▶ SAFe on saanut osakseen myös paljon kritiikkiä
  - ▶ Osa kritiikistä kohdistuu SAFen määrittelemän prosessin raskauteen
  - ▶ Osa taas SAFe:n top down -management luonteseen
- ▶ Ken Schwaber on kyseenalaistanut onko SAFe ylipäättäään ketterä menetelmä
  - ▶ *Individuals and Interactions Over Processes and Tools*
  - ▶ SAFe taas prosessina vaikuttaa kovin raskaalta

# LeSS eli Large Scale Scrum

## LeSS eli Large Scale Scrum

- ▶ LeSS:in taustalla on Craig Larman ja Bas Vodde
  - ▶ konsultteina 2000-luvun alussa Nokia Siemens Networksilla

# LeSS eli Large Scale Scrum

- ▶ LeSS:in taustalla on Craig Larman ja Bas Vodde
  - ▶ konsultteina 2000-luvun alussa Nokia Siemens Networksilla
- ▶ Erittäin yksinkertainen, vahvasti Scrumiin pohjautuva
  - ▶ Uusia rooleja, artifakteja ja palavereja ei ole

# LeSS eli Large Scale Scrum

- ▶ LeSS:in taustalla on Craig Larman ja Bas Vodde
  - ▶ konsultteina 2000-luvun alussa Nokia Siemens Networksilla
- ▶ Erittäin yksinkertainen, vahvasti Scrumiin pohjautuva
  - ▶ Uusia rooleja, artifakteja ja palavereja ei ole
- ▶ Kaksi eri versiota
  - ▶ *LeSS* tilanteisiin, missä tuotetta tekee 2-8 scrum-tiimiä
  - ▶ *LeSS Huge* tilanteisiin, missä tiimejä tarvitaan suurempi määrä

## LeSSin perusperiaatteet

- ▶ Sekä LeSS että LeSS Huge perustuvat seuraavaan

## LeSSin perusperiaatteet

- ▶ Sekä LeSS että LeSS Huge perustuvat seuraavaan
- ▶ Kehitetään yhtä tuotetta, jolla on yksi product owner ja yksi product backlog

## LeSSin perusperiaatteet

- ▶ Sekä LeSS että LeSS Huge perustuvat seuraavaan
- ▶ Kehitetään yhtä tuotetta, jolla on yksi product owner ja yksi product backlog
- ▶ Kaikilla tiimeillä on samaan aikaan etenevät sprintit

## LeSSin perusperiaatteet

- ▶ Sekä LeSS että LeSS Huge perustuvat seuraavaan
- ▶ Kehitetään yhtä tuotetta, jolla on yksi product owner ja yksi product backlog
- ▶ Kaikilla tiimeillä on samaan aikaan etenevät sprintit
- ▶ Tiimit tekevät sprintin aikana yhdessä tuotteesta uuden version
  - ▶ *one shippable product increment*

## LeSSin perusperiaatteet

- ▶ Sekä LeSS että LeSS Huge perustuvat seuraavaan
- ▶ Kehitetään yhtä tuotetta, jolla on yksi product owner ja yksi product backlog
- ▶ Kaikilla tiimeillä on samaan aikaan etenevät sprintit
- ▶ Tiimit tekevät sprintin aikana yhdessä tuotteesta uuden version
  - ▶ *one shippable product increment*
- ▶ Tiimit ovat
  - ▶ cross functional
  - ▶ cross component, full stack
  - ▶ feature teams

## LeSSin perusperiaatteet

- ▶ Jos yrityksellä on useita tuotteita, niitä kutakin varten on oma LeSS-toteutuksensa
  - ▶ LeSS ei ota kantaa siihen miten firma hallinnoi tuoteperheitään

## LeSSin perusperiaatteet

- ▶ Jos yrityksellä on useita tuotteita, niitä kutakin varten on oma LeSS-toteutuksensa
  - ▶ LeSS ei ota kantaa siihen miten firma hallinnoi tuoteperheitään
- ▶ LeSS korostaa että kyse *ei ole* Scrumin päälle rakennettu erillinen kerros

## LeSSin perusperiaatteet

- ▶ Jos yrityksellä on useita tuotteita, niitä kutakin varten on oma LeSS-toteutuksensa
  - ▶ LeSS ei ota kantaa siihen miten firma hallinnoi tuoteperheitään
- ▶ LeSS korostaa että kyse *ei ole* Scrumin päälle rakennettu erillinen kerros
- ▶ vaan ideana on soveluttaa Scrumin elementtejä laajemmassa skaalassa
  - ▶ LeSS is “barely sufficient methodology”

## More with less

- ▶ Periaatteet ovat lähes samat kuin SAFe:ssa, yksi periaatteesta tekee kuitenkin selvää eroa menetelmien välille

## More with less

- ▶ Periaatteet ovat lähes samat kuin SAFe:ssa, yksi periaatteesta tekee kuitenkin selvää eroa menetelmien välille
- ▶ We don't want
  - ▶ **more roles** because more roles leads to less responsibility to teams
  - ▶ **more artifacts** because more artifacts leads to a greater distance between teams and customers
  - ▶ **more process** because that leads to less learning and team ownership of process

## More with less

- ▶ Periaatteet ovat lähes samat kuin SAFe:ssa, yksi periaatteesta tekee kuitenkin selvää eroa menetelmien välille
- ▶ We don't want
  - ▶ **more roles** because more roles leads to less responsibility to teams
  - ▶ **more artifacts** because more artifacts leads to a greater distance between teams and customers
  - ▶ **more process** because that leads to less learning and team ownership of process
- ▶ Instead we want
  - ▶ more **responsible teams** by having less roles
  - ▶ more **customer-focused teams** building useful products by having less artifacts
  - ▶ more **team ownership of process** and more meaningful work by having less defined processes

## More with less

- ▶ Periaatteet ovat lähes samat kuin SAFe:ssa, yksi periaatteesta tekee kuitenkin selvää eroa menetelmien välille
- ▶ We don't want
  - ▶ **more roles** because more roles leads to less responsibility to teams
  - ▶ **more artifacts** because more artifacts leads to a greater distance between teams and customers
  - ▶ **more process** because that leads to less learning and team ownership of process
- ▶ Instead we want
  - ▶ more **responsible teams** by having less roles
  - ▶ more **customer-focused teams** building useful products by having less artifacts
  - ▶ more **team ownership of process** and more meaningful work by having less defined processes
- ▶ **We want more with less**

# LeSS

- ▶ Katsotaan hieman tarkemmin LeSS:in pienempää konfiguraatiota

# LeSS

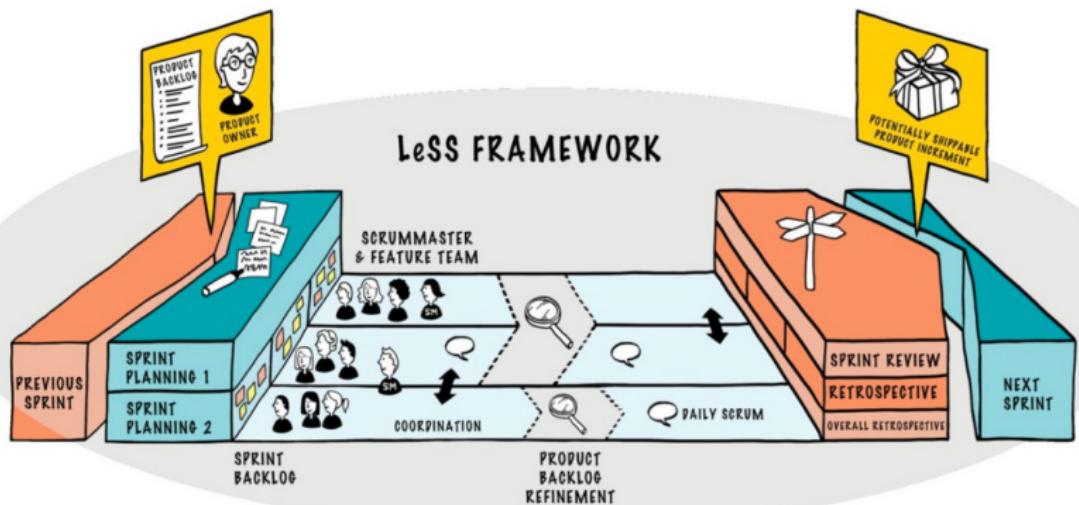
- ▶ Katsotaan hieman tarkemmin LeSS:in pienempää konfiguraatiota
- ▶ Roolit
  - ▶ yksi product owner
  - ▶ 2-8 tiimiä
  - ▶ yksi scrum master 1-3 tiimiä kohti

# LeSS

- ▶ Katsotaan hieman tarkemmin LeSS:in pienempää konfiguraatiota
- ▶ Roolit
  - ▶ yksi product owner
  - ▶ 2-8 tiimiä
  - ▶ yksi scrum master 1-3 tiimiä kohti
- ▶ Tiimit:
  - ▶ cross functional
  - ▶ cross component / full stack
  - ▶ työskentelevät saman koodin parissa

- ▶ Artefaktit
  - ▶ yksi product backlog
  - ▶ yhteinen kehitettävä tuote
  - ▶ tiimikohtaiset sprinttibacklogit

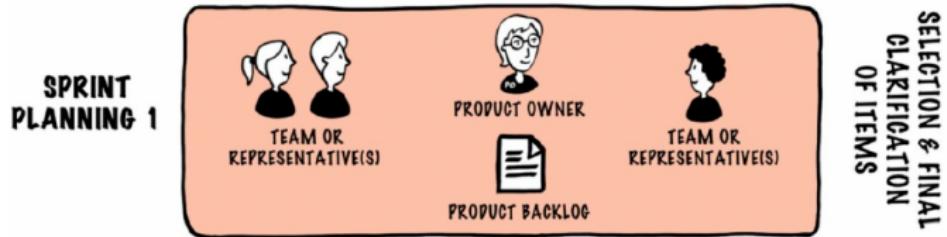
- ▶ Artefaktit
  - ▶ yksi product backlog
  - ▶ yhteinen kehitettävä tuote
  - ▶ tiimikohtaiset sprinttibacklogit
- ▶ Kaikille yhteinen sprintti



# LeSS: kaksiosainen sprintin suunnittelu

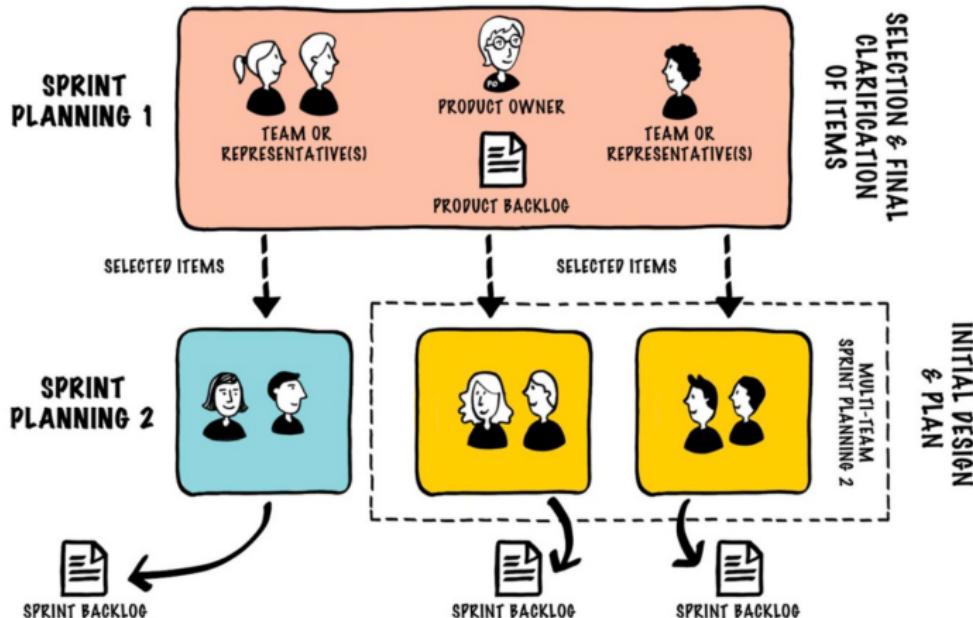
# LeSS: kaksiosainen sprintin suunnittelu

- ▶ Ensimmäisessä osassa product owner ja tiimien edustajat valitsevat backlogilta tiimeille seuraavan sprintin storyt



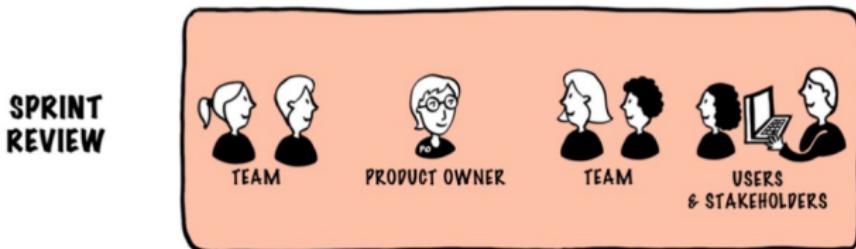
# LeSS: kaksiosainen sprintin suunnittelu

- ▶ Ensimmäisessä osassa product owner ja tiimien edustajat valitsevat backlogilta tiimeille seuraavan sprintin storyt

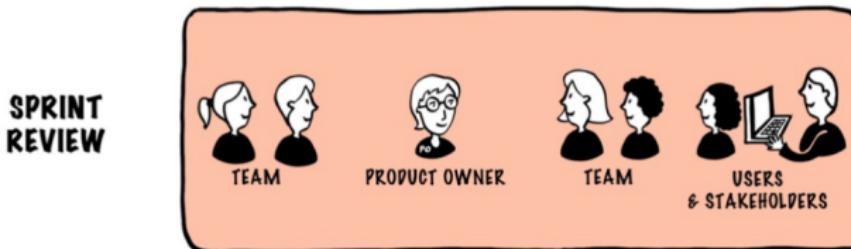


- ▶ Toisessa osassa tiimit muodostavat omat sprint backlogit

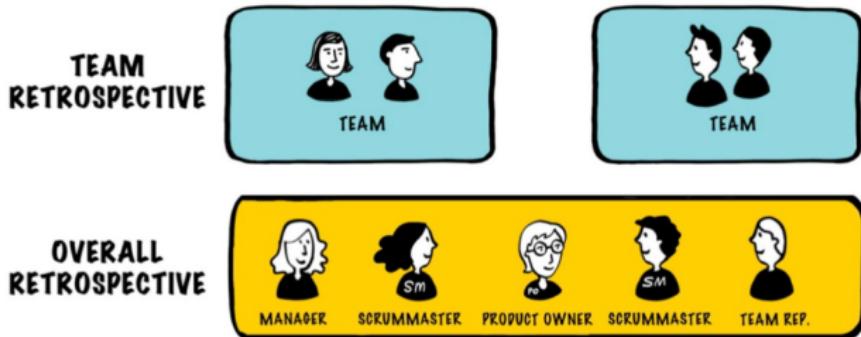
- ▶ Kaikkien tiimien yhteinen aikaansaannos (one shippable product increment) katselmoidaan yhdessä



- ▶ Kaikkien tiimien yhteinen aikaansaannos (one shippable product increment) katselmoidaan yhdessä



- ▶ Retrospektiivi on kaksitasoinen
  - ▶ tiimikohtainen
  - ▶ overall-retrospektiivi: edustus kaikista tiimeistä ja mahdollisesti yrityksen johdosta



## LeSS: Muu tiimien välinen koordinointi

- ▶ Yhteisen sprintin suunnittelun, reviewin ja overall-retrospektiivin lisäksi ei edellytä muita yhteisiä tapaamisia

## LeSS: Muu tiimien välinen koordinointi

- ▶ Yhteisen sprintin suunnittelun, reviewin ja overall-retrospektiivin lisäksi ei edellytä muita yhteisiä tapaamisia
- ▶ LeSS antaa joukon aiheeseen liittyviä ohjeita ja suosituksia

## LeSS: Muu tiimien välinen koordinointi

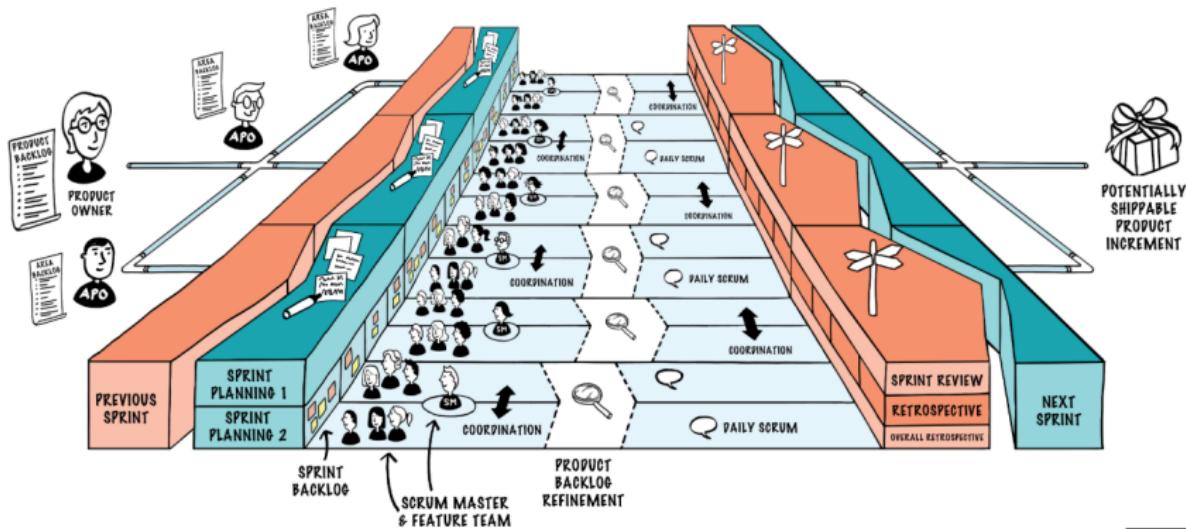
- ▶ Yhteisen sprintin suunnittelun, reviewin ja overall-retrospektiivin lisäksi ei edellytä muita yhteisiä tapaamisia
- ▶ LeSS antaa joukon aiheeseen liittyviä ohjeita ja suosituksia
- ▶ Tiimit päättävät keskenään miten tiimien välinen koordinointi tapahtuu
  - ▶ Emphasize Just Talk and informal networks
  - ▶ communicate in code
  - ▶ cross-team meetings
  - ▶ component mentors
  - ▶ open spaces
  - ▶ scouts

## LeSS: Muu tiimien välinen koordinointi

- ▶ Yhteisen sprintin suunnittelun, reviewin ja overall-retrospektiivin lisäksi ei edellytä muita yhteisiä tapaamisia
- ▶ LeSS antaa joukon aiheeseen liittyviä ohjeita ja suosituksia
- ▶ Tiimit päättävät keskenään miten tiimien välinen koordinointi tapahtuu
  - ▶ Emphasize Just Talk and informal networks
  - ▶ communicate in code
  - ▶ cross-team meetings
  - ▶ component mentors
  - ▶ open spaces
  - ▶ scouts
- ▶ Scrum of Scrums -palaverit mainitaan, mutta suositellaan informaalimpia kommunikaation muotoja

# LeSS huge

- ▶ Yksi tuote, backlog ja vastuunalainen product owner
- ▶ Backlog jaetaan *vaatimusalueisiin* (requirement area)
  - ▶ jokaiselle alueelle siitä vastaava *area product owner*
  - ▶ muodostavat product owner -tiimi
  - ▶ backlogin aluekohtaiset näkymät



## LeSS vs SAFe

- ▶ SAFen ja LeSSin juuret Suomessa ja Nokialla

## LeSS vs SAFe

- ▶ SAFen ja LeSSin juuret Suomessa ja Nokialla
- ▶ Nokian organisaatorakenteen takia Nokia Mobile Phonesin (NMP) ja Nokia Siemens Networksin (NSN) ohjelmistokehitystapa oli täysin erilainen

## LeSS vs SAFe

- ▶ SAFen ja LeSSin juuret Suomessa ja Nokialla
- ▶ Nokian organisaatorakenteen takia Nokia Mobile Phonesin (NMP) ja Nokia Siemens Networksin (NSN) ohjelmistokehitystapa oli täysin erilainen
- ▶ SAFe (NMP) ja LeSS (NSN) ovat samoista taustaperiaatteistaan ja yhtesisestä syntykonsernistä huolimatta hyvin erilaisia menetelmiä

## LeSS vs SAFe

- ▶ SAFen ja LeSSin juuret Suomessa ja Nokialla
- ▶ Nokian organisaatorakenteen takia Nokia Mobile Phonesin (NMP) ja Nokia Siemens Networksin (NSN) ohjelmistokehitystapa oli täysin erilainen
- ▶ SAFe (NMP) ja LeSS (NSN) ovat samoista taustaperiaatteistaan ja yhtesisestä syntykonsernistä huolimatta hyvin erilaisia menetelmiä
- ▶ SAFe suosittu yritysjohdon keskuudessa, mutta saanut paljon kriitikkiä
  - ▶ Ohjelmistokehittäjiltä en ole kuullut SAFesta juurikaan kiitosta

## LeSS vs SAFe

- ▶ SAFen ja LeSSin juuret Suomessa ja Nokialla
- ▶ Nokian organisaatorakenteen takia Nokia Mobile Phonesin (NMP) ja Nokia Siemens Networksin (NSN) ohjelmistokehitystapa oli täysin erilainen
- ▶ SAFe (NMP) ja LeSS (NSN) ovat samoista taustaperiaatteistaan ja yhtesisestä syntykonsernista huolimatta hyvin erilaisia menetelmiä
- ▶ SAFe suosittu yritysjohdon keskuudessa, mutta saanut paljon kritiikkiä
  - ▶ Ohjelmistokehittäjiltä en ole kuullut SAFesta juurikaan kiitosta
- ▶ SAFe:n kotia Nokia Mobile Phonesia ei enää ole
- ▶ Nokia Networks taas on nykyinen Nokia ja soveltaa yhä LeSS-menetelmää

# LeSS vs SAFe

- ▶ SAFen ja LeSSin juuret Suomessa ja Nokialla
- ▶ Nokian organisaatorakenteen takia Nokia Mobile Phonesin (NMP) ja Nokia Siemens Networksin (NSN) ohjelmistokehitystapa oli täysin erilainen
- ▶ SAFe (NMP) ja LeSS (NSN) ovat samoista taustaperiaatteistaan ja yhtesisestä syntykonsernista huolimatta hyvin erilaisia menetelmiä
- ▶ SAFe suosittu yritysjohdon keskuudessa, mutta saanut paljon kritiikkiä
  - ▶ Ohjelmistokehittäjiltä en ole kuullut SAFesta juurikaan kiitosta
- ▶ SAFe:n kotia Nokia Mobile Phonesia ei enää ole
- ▶ Nokia Networks taas on nykyinen Nokia ja soveltaa yhä LeSS-menetelmää
- ▶ SAFe:n asema vahva, ja vahvistuu koko ajan

TAUKO 10 min

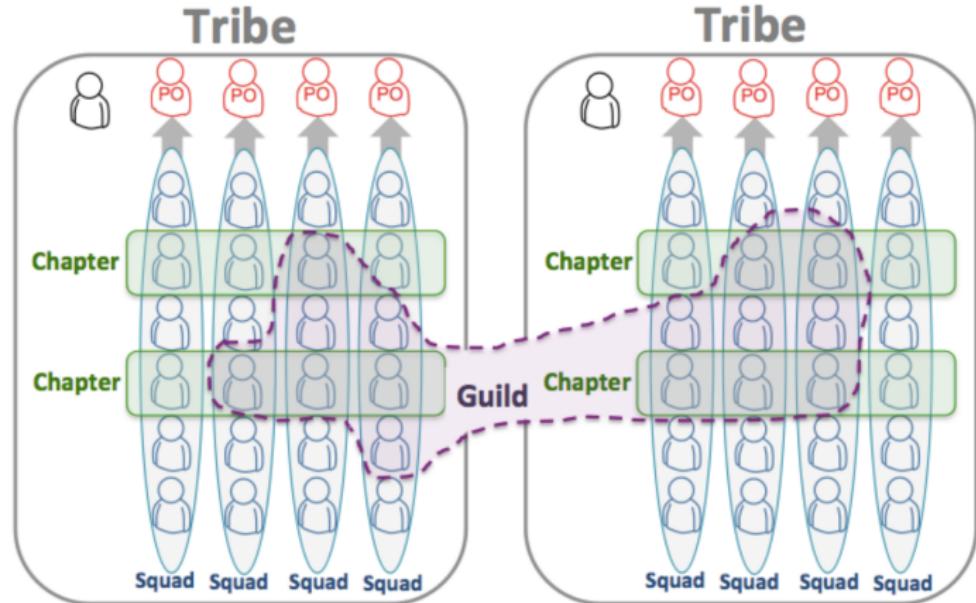
# Spotifyn skaalaamisen malli

## Spotifyn skaalaamisen malli

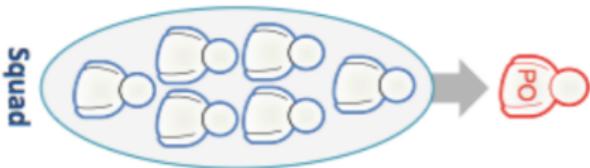
- ▶ 2012 julkaistu artikkeli *Scaling agile @ Spotify* saavutti paljon huomiota
- ▶ Miten Spotify skaalasi toimintansa muutamassa vuodessa yhdestä useampaan kymmeneen tiimiin

# Spotifyn skaalaamisen malli

- ▶ 2012 julkaistu artikkeli *Scaling agile @ Spotify* saavutti paljon huomiota
- ▶ Miten Spotify skaalasi toimintansa muutamassa vuodessa yhdestä useampaan kymmeneen tiimiin

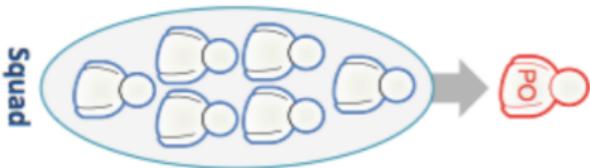


## Squad eli tiimi



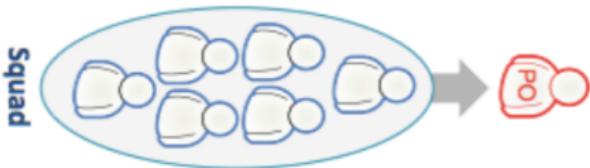
- ▶ Itseorganisoituva, cross-functional, featuretiimi
  - ▶ työskentelee yhdessä tilassa

## Squad eli tiimi



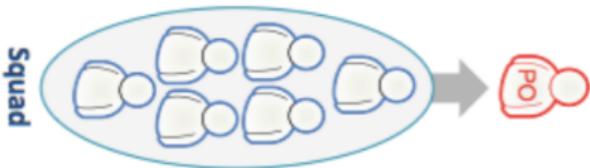
- ▶ Itseorganisoituva, cross-functional, featuretiimi
  - ▶ työskentelee yhdessä tilassa
- ▶ Saa itse päättää prosessinsa
  - ▶ ei välittämättä Scrum

# Squad eli tiimi



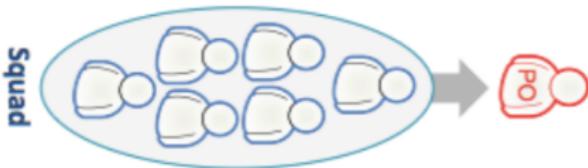
- ▶ Itseorganisoituva, cross-functional, featuretiimi
  - ▶ työskentelee yhdessä tilassa
- ▶ Saa itse päättää prosessinsa
  - ▶ ei välittämättä Scrum
- ▶ Tiimeillä vahva liiketoiminnallinen missio
  - ▶ suuri autonomia myös liiketoiminnallisiin päätöksiin
  - ▶ mahdollisimman suora asiakaskontakti
  - ▶ firman vision puitteissa

# Squad eli tiimi



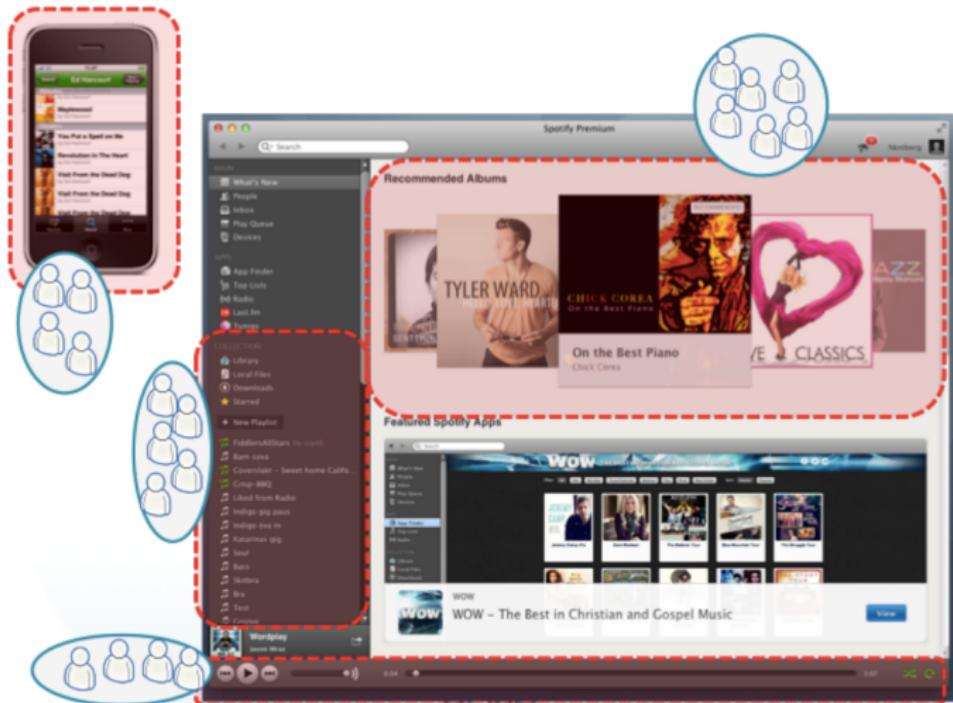
- ▶ Itseorganisoitava, cross-functional, featuretiimi
  - ▶ työskentelee yhdessä tilassa
- ▶ Saa itse päättää prosessinsa
  - ▶ ei välittämättä Scrum
- ▶ Tiimeillä vahva liiketoiminnallinen missio
  - ▶ suuri autonomia myös liiketoiminnalliisiin päätöksiin
  - ▶ mahdollisimman suora asiakaskontakti
  - ▶ firman vision puitteissa
- ▶ Ministartup

# Squad eli tiimi



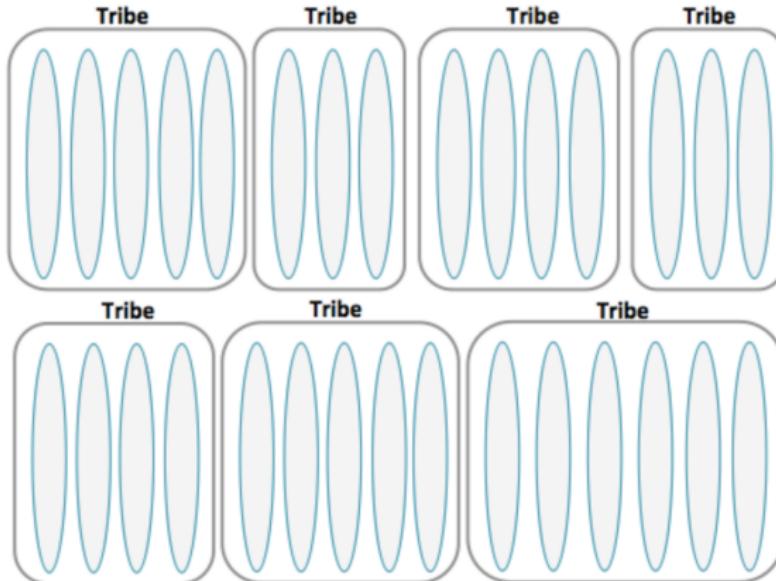
- ▶ Itseorganisoituva, cross-functional, featuretiimi
  - ▶ työskentelee yhdessä tilassa
- ▶ Saa itse päättää prosessinsa
  - ▶ ei välittämättä Scrum
- ▶ Tiimeillä vahva liiketoiminnallinen missio
  - ▶ suuri autonomia myös liiketoiminnallisiin päätöksiin
  - ▶ mahdollisimman suora asiakaskontakti
  - ▶ firman vision puitteissa
- ▶ Ministartup
- ▶ Agile coachit apuna

- Tiimi vastaa usein suoraan asiakkaalle näkyvästä toiminnallisuudesta



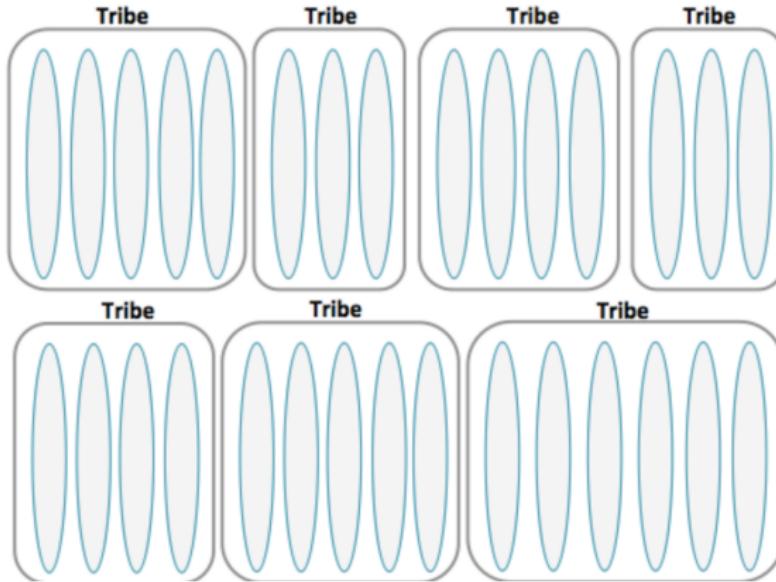
- Tai muita tiimejä palvelevasta Spotifyn sisäisestä “palvelusta”

# Tribe eli heimo



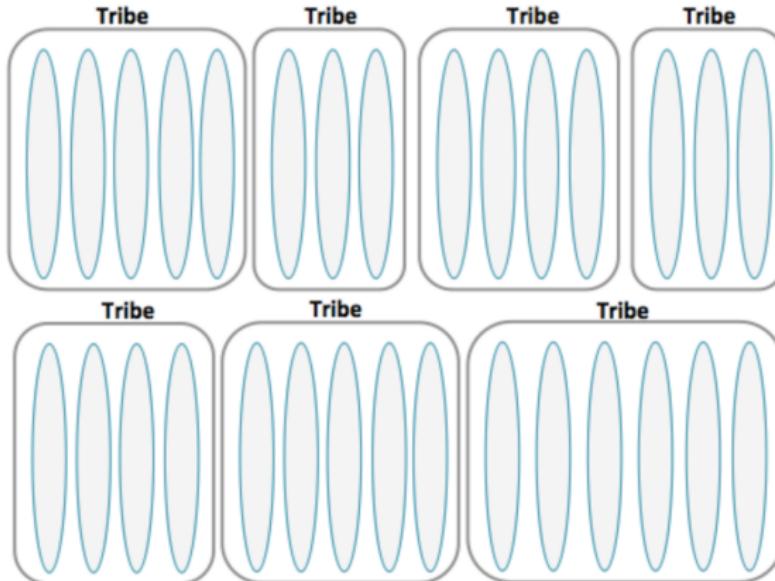
- ▶ Joukko tiimejä jotka työskentelevät jonkin loogisen kokonaisuuden parissa

# Tribe eli heimo



- ▶ Joukko tiimejä jotka työskentelevät jonkin loogisen kokonaisuuden parissa
- ▶ Sijoitettu fyysisesti lähekkäin

# Tribe eli heimo



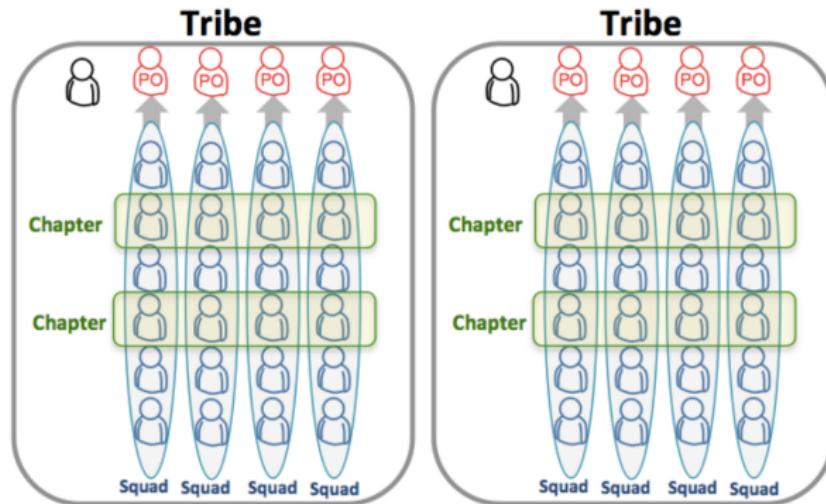
- ▶ Joukko tiimejä jotka työskentelevät jonkin loogisen kokonaisuuden parissa
- ▶ Sijoitettu fyysisesti lähekkäin
- ▶ Yhteisiä kokousia: suunnittelua, demoja...

## Chapter eli jaos

- ▶ Riskinä että autonomiset tiimit joutuvat kohtaamaan toistuvasti samoja ongelmia

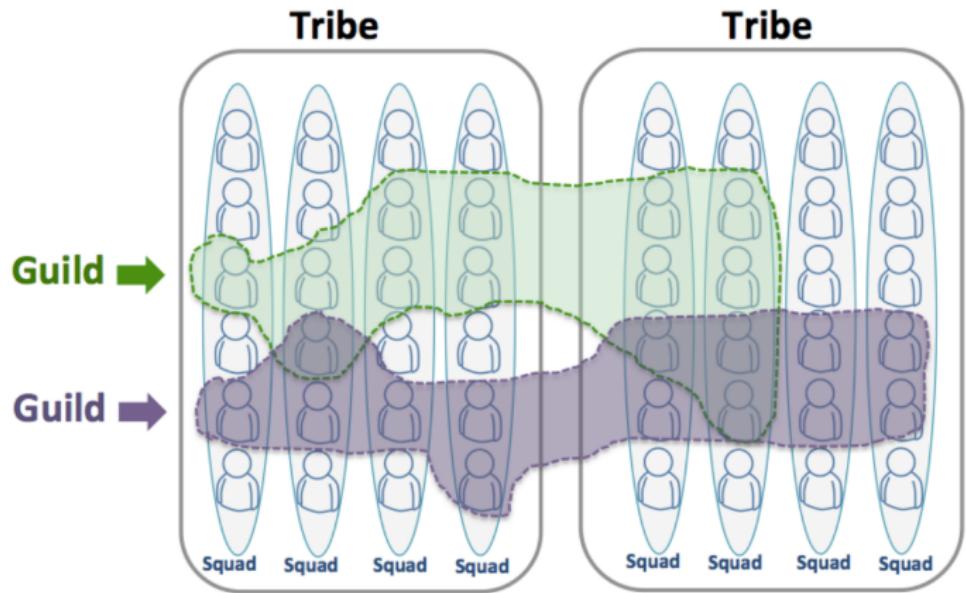
# Chapter eli jaos

- ▶ Riskinä että autonomiset tiimit joutuvat kohtaamaan toistuvasti samoja ongelmia



- ▶ Heimon eri tiimien samaa asiaa tekevien ihmisten joukko
  - ▶ esim. testaajat, designerit, frontend-kehittäjät
- ▶ levittää tietoa tiimistä toiseen

# Guild eli kilta



- ▶ Yli heimorajojen ylittyvä samasta asiasta kiinnostuneiden yhteisö

## Spotifyn “malli”

- ▶ Kertoo siitä mikä tilanne Spotifyllä noin 2012-14

## Spotifyn “malli”

- ▶ Kertoo siitä mikä tilanne Spotifyllä noin 2012-14
- ▶ Dokumentoi ainoastaan pintapuolisesti tiettyjäasioita siitä miten firma oli organisoitunut
  - ▶ ei tarkoitettu ohjeeksi muille ...

## Spotifyn “malli”

- ▶ Kertoo siitä mikä tilanne Spotifyllä noin 2012-14
- ▶ Dokumentoi ainoastaan pintapuolisesti tiettyjäasioita siitä miten firma oli organisoitunut
  - ▶ ei tarkoitettu ohjeeksi muille ...
- ▶ Monet asiat ovat Spotifyllä sittemmin muuttuneet
  - ▶ “Spotify ei käytä Spotifyn mallia”

## Spotifyn “malli”

- ▶ Kertoo siitä mikä tilanne Spotifyllä noin 2012-14
- ▶ Dokumentoi ainoastaan pintapuolisesti tiettyjäasioita siitä miten firma oli organisoitunut
  - ▶ ei tarkoitettu ohjeeksi muille ...
- ▶ Monet asiat ovat Spotifyllä sittemmin muuttuneet
  - ▶ “Spotify ei käytä Spotifyn mallia”
- ▶ Levisi yllättäen maailmalle
  - ▶ myös Suomessa ruvettiin pöhisemään heimoista

## Spotifyn “malli”

- ▶ Kertoo siitä mikä tilanne Spotifyllä noin 2012-14
- ▶ Dokumentoi ainoastaan pintapuolisesti tiettyjäasioita siitä miten firma oli organisoitunut
  - ▶ ei tarkoitettu ohjeeksi muille ...
- ▶ Monet asiat ovat Spotifyllä sittemmin muuttuneet
  - ▶ “Spotify ei käytä Spotifyn mallia”
- ▶ Levisi yllättäen maailmalle
  - ▶ myös Suomessa ruvettiin pöhisemään heimoista
- ▶ Saanut paljon kritiikkiä
  - ▶ “keksinyt” tunnettuja asioita uudelleen uusilla termeillä
  - ▶ Spotifyn sisältä (antaa utopistisen kuvan firman toiminnasta)
  - ▶ myös mallin soveltamista ilman sen hengen ymmärtämistä arvosteltu

# Playlist: Spotifyn “tarina” stiimauspalveluissa...

IMDb Menu All Search IMDb IMDbPro Watchlist Sign In EN ▾

Episode guide 6 > Cast & crew · User reviews · Trivia All topics Cast & crew User reviews Trivia All topics

**The Playlist**

TV Mini Series · 2022 · TV-MA · 5h

IMDb RATING YOUR RATING POPULARITY

7.5/10 7K Rate 515 166

Play trailer 1:38

13 VIDEOS

17 PHOTOS

Biography Drama Music + Add to Watchlist

Miten laajalti ketterää/leania käytetään

## Miten laajalti ketterää/leania käytetään

- ▶ Paljon kyselytutkimuksia, luvut vaihtelevat 46-86 % välillä
  - ▶ Project management institute (2018): 46 %
  - ▶ Stack overflow (yli 200000 vastaajaa, 2018): 85.9%

## Miten laajalti ketterää/leania käytetään

- ▶ Paljon kyselytutkimuksia, luvut vaihtelevat 46-86 % välillä
  - ▶ Project management institute (2018): 46 %
  - ▶ Stack overflow (yli 200000 vastaajaa, 2018): 85.9%
- ▶ Akateemisia tutkimuksia:

## Miten laajalti ketterää/leania käytetään

- ▶ Paljon kyselytutkimuksia, luvut vaihtelevat 46-86 % välillä
  - ▶ Project management institute (2018): 46 %
  - ▶ Stack overflow (yli 200000 vastaajaa, 2018): 85.9%
- ▶ Akateemisia tutkimuksia:
- ▶ Oulun yliopisto *Survey on Agile and Lean usage in Finnish software industry* 2012:
  - ▶ 58% vastanneista 200 yrityksestä käytti agilea tai leania

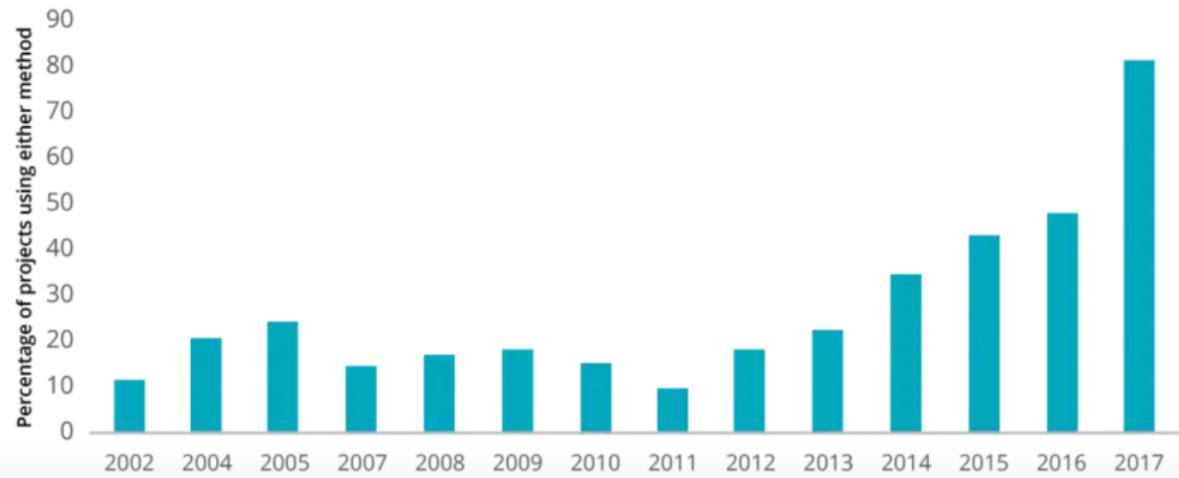
## Miten laajalti ketterää/leania käytetään

- ▶ Paljon kyselytutkimuksia, luvut vaihtelevat 46-86 % välillä
  - ▶ Project management institute (2018): 46 %
  - ▶ Stack overflow (yli 200000 vastaajaa, 2018): 85.9%
- ▶ Akateemisia tutkimuksia:
- ▶ Oulun yliopisto *Survey on Agile and Lean usage in Finnish software industry* 2012:
  - ▶ 58% vastanneista 200 yrityksestä käytti agilea tai leania
- ▶ Turun yliopisto ym *Adoption and Suitability of Software Development Methods and Practices* 2016:
  - ▶ Scrum 71.2%
  - ▶ Kanban 49.5%
  - ▶ Lean 39.7%
  - ▶ Vesiputous 35.3%

► Helsingin yliopiston ja Nitorin loppuvuodesta 2018 tekemän selvitys



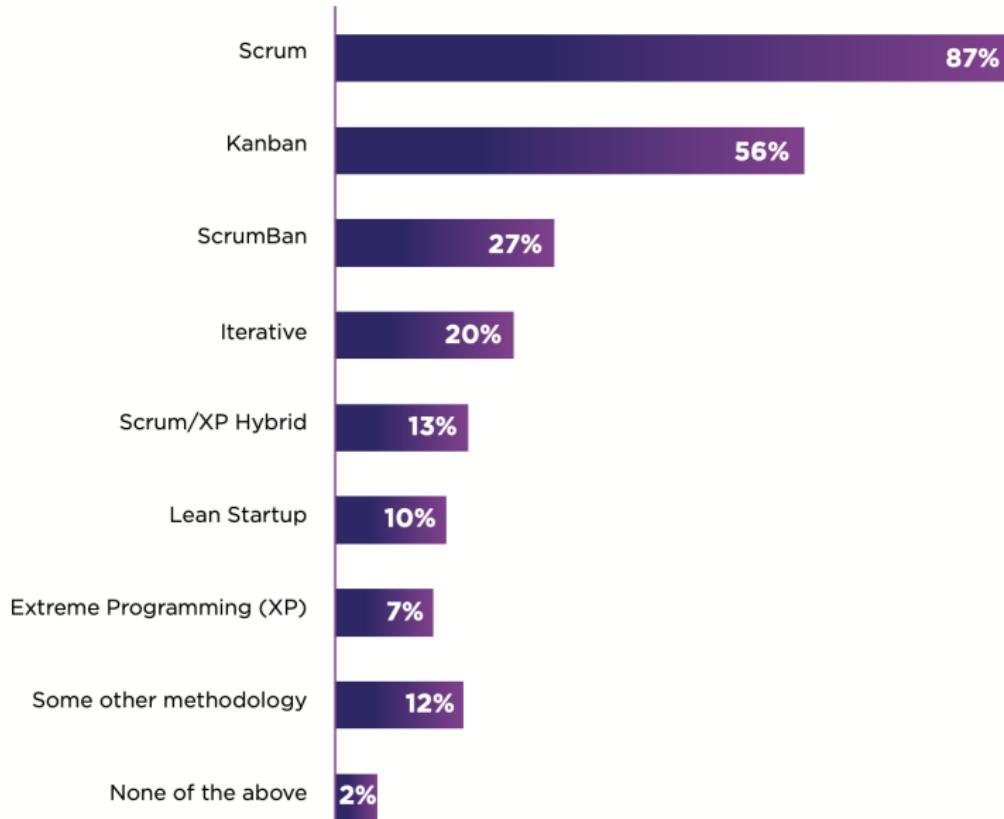
► Yhdysvaltojen hallituksen alaiset ohjelmistoprojektit



# State of Agile -raportti: vuodesta 2005, uusin 2022

# State of Agile -raportti: vuodesta 2005, uusin 2022

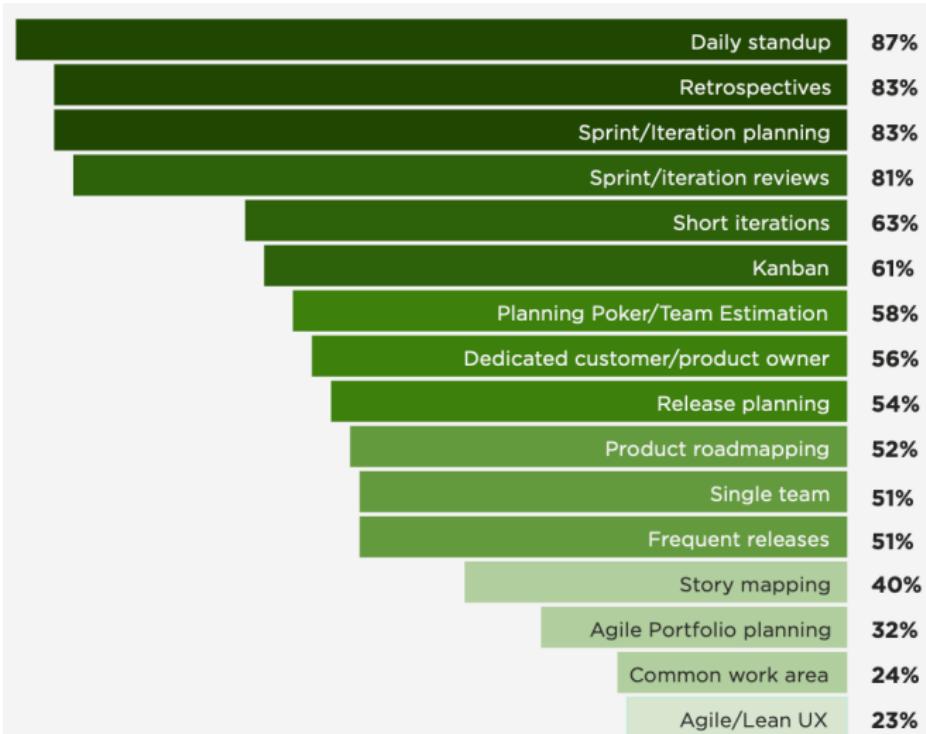
## ► Scrum dominoi



## ► Projektinhallintakäytänteet (2021 raportista)

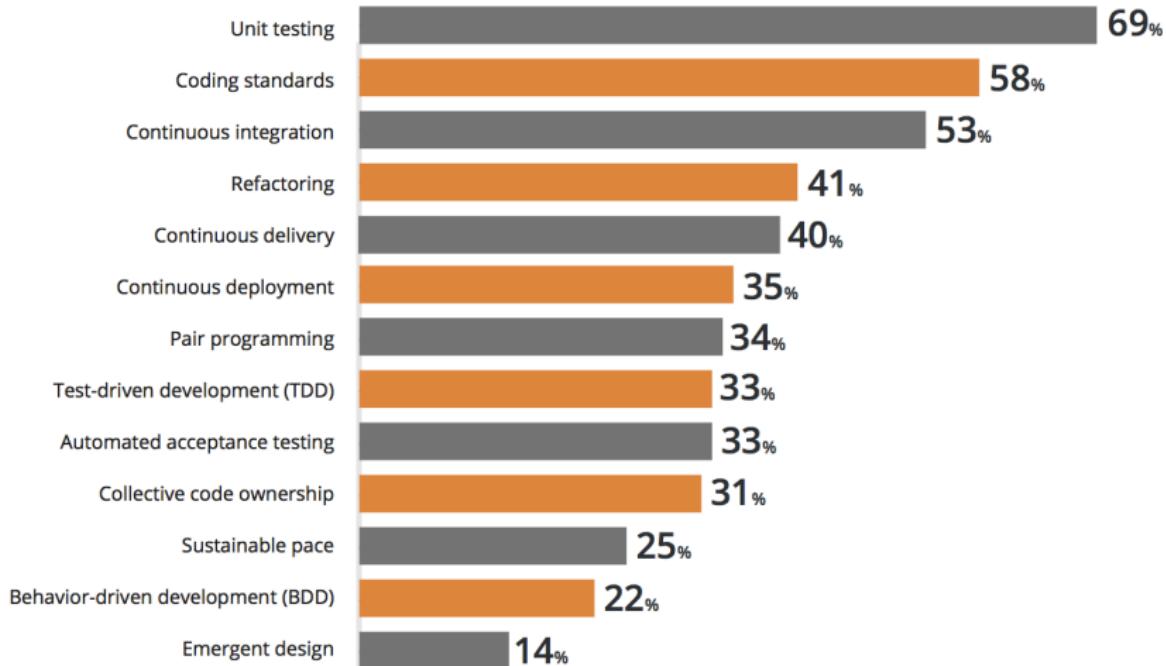


## ► Projektinhallintakäytänteet (2021 raportista)

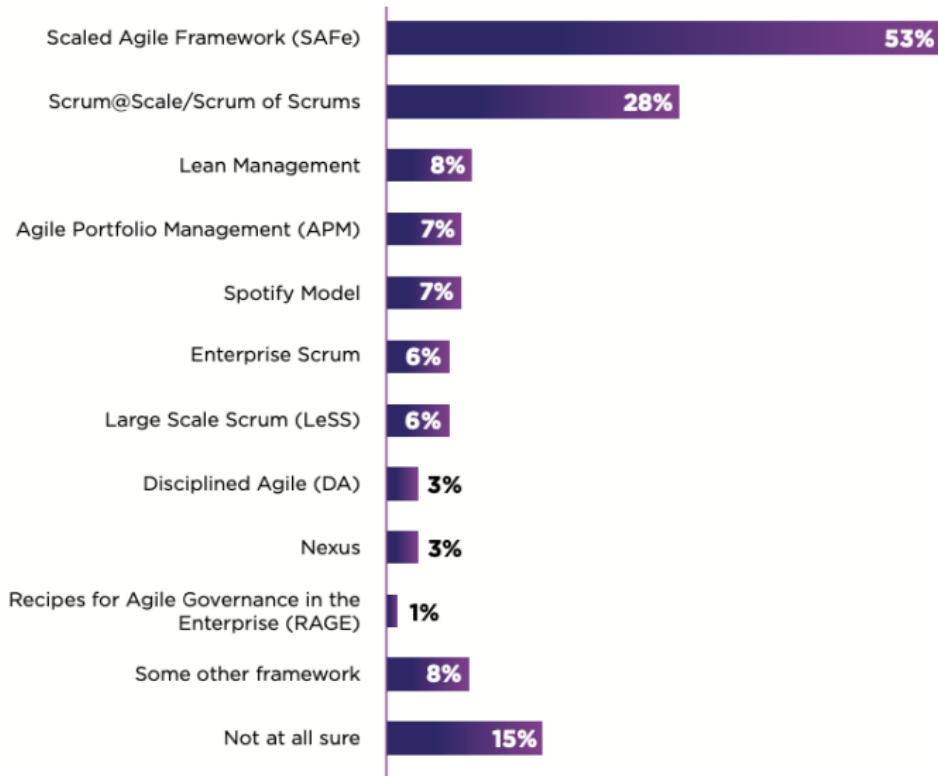


► short iterations 63%

## ► Tekniset käytänteet (2021 raportista)

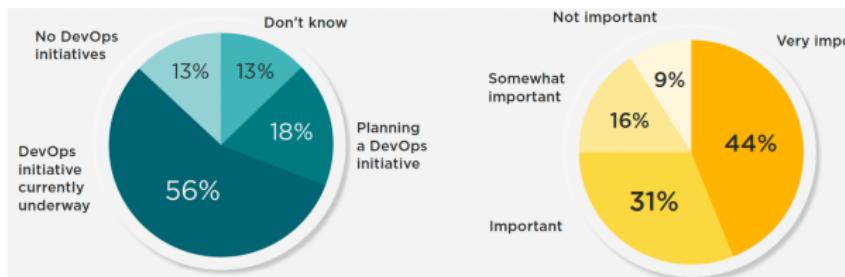


# Laajan mittakaavan ketterä



# DevOps

- ▶ onko tiheään julkaisuun yms pyrkivä DevOps-“kulttuuri” käytössä ja piedetäänkö sitä tärkeänä?



## Toimiiko ketterä ohjelmistokehitys

- ▶ Standish groupin *Chaos rapport*, vuodesta 1995
  - ▶ Ohjelmistoprojektien onnistuminen

# Toimiiko ketterä ohjelmistokehitys

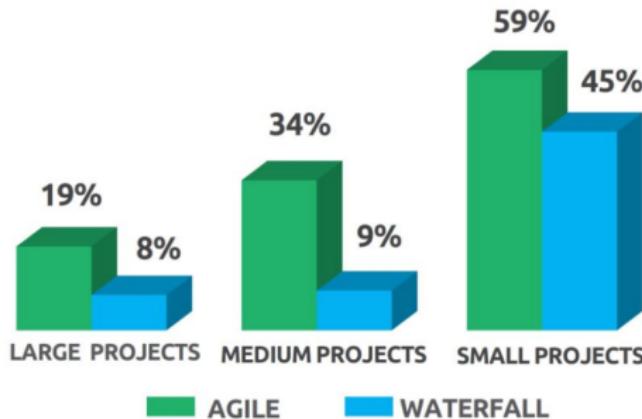
- ▶ Standish groupin *Chaos rapport*, vuodesta 1995
  - ▶ Ohjelmistoprojektien onnistuminen
- ▶ 2020:

## PROJECT SUCCESS RATES AGILE VS WATERFALL

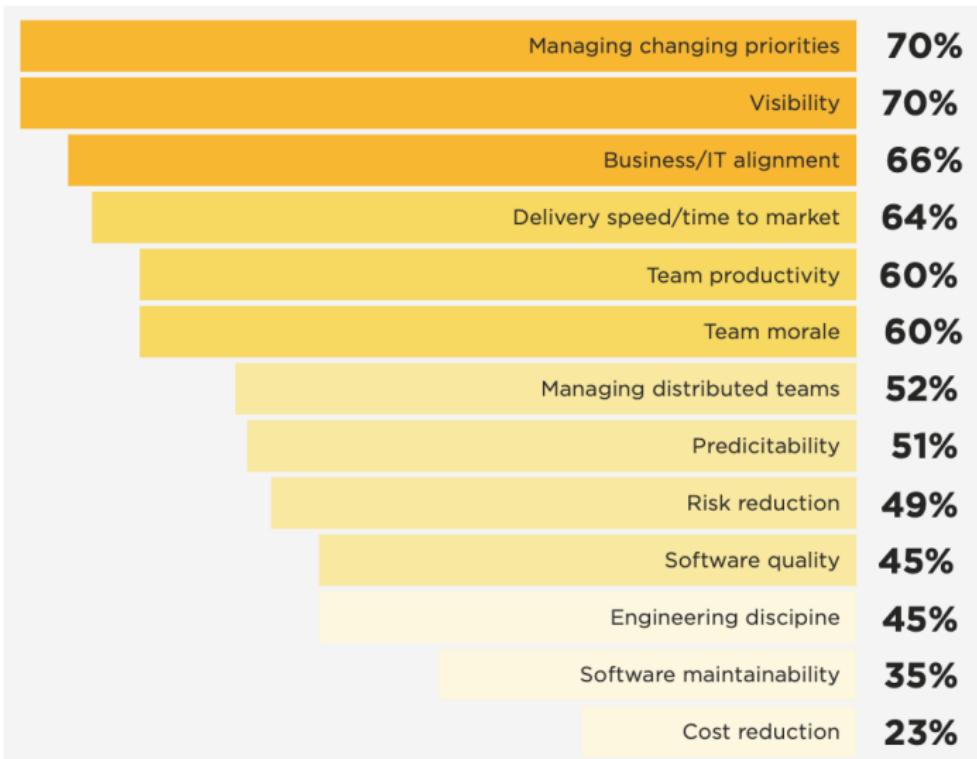


- ▶ successful: aikataulussa, budjetissa, sovittu toiminnallisuus

- ▶ Projektin koolla on todella suuri merkitys onnistumisen kannalta.



► Ketteryydellä saavutettuja hyötyjä



## Toimiiko ketterä

- ▶ Evidenssiä on, mutta...

## Toimiiko ketterä

- ▶ Evidenssiä on, mutta...
- ▶ Kaikki edelliset olivat kyselytutkimuksia
  - ▶ käsitteitä ei ole kunnolla määritelty (Scrum vs Scrumbut)
  - ▶ osallistuneet eivät ehkä edusta tasaisesti koko populaatiota
  - ▶ kyselyjen tekijät eivät ole puolueettomia menetelmien suhteen

## Toimiiko ketterä

- ▶ Evidenssiä on, mutta...
- ▶ Kaikki edelliset olivat kyselytutkimuksia
  - ▶ käsitteitä ei ole kunnolla määritelty (Scrum vs Scrumbut)
  - ▶ osallistuneet eivät ehkä edusta tasaisesti koko populaatiota
  - ▶ kyselyjen tekijät eivät ole puolueettomia menetelmien suhteen
- ▶ Akateemisenkin tutkimuksen laatu ja tulosten yleistettävyys vaihtelee

## Toimiiko ketterä

- ▶ Evidenssiä on, mutta...
- ▶ Kaikki edelliset olivat kyselytutkimuksia
  - ▶ käsitteitä ei ole kunnolla määritelty (Scrum vs Scrumbut)
  - ▶ osallistuneet eivät ehkä edusta tasaisesti koko populaatiota
  - ▶ kyselyjen tekijät eivät ole puolueettomia menetelmien suhteen
- ▶ Akateemisenkin tutkimuksen laatu ja tulosten yleistettävyys vaihtelee
- ▶ Ohjelmistokehityksessä liian paljon muuttujia, jotta jonkin yksittäisen tekijän vaikutusta voitaisiin mitata empiirisesti

## Toimiiko ketterä

- ▶ Evidenssiä on, mutta...
- ▶ Kaikki edelliset olivat kyselytutkimuksia
  - ▶ käsitteitä ei ole kunnolla määritelty (Scrum vs Scrumbut)
  - ▶ osallistuneet eivät ehkä edusta tasaisesti koko populaatiota
  - ▶ kyselyjen tekijät eivät ole puolueettomia menetelmien suhteen
- ▶ Akateemisenkin tutkimuksen laatu ja tulosten yleistettävyys vaihtelee
- ▶ Ohjelmistokehityksessä liian paljon muuttujia, jotta jonkin yksittäisen tekijän vaikutusta voitaisiin mitata empiirisesti
- ▶ Menetelmiä soveltavat ihmiset, ja mittaustulos yhdellä tiimillä ei vältämättä yleisty muihin olosuhteisiin

## Ketterä post covid?

- ▶ eräs ketterän perusolettamuksia on se, että tiimi jakaa yhteisen työtilan
  - ▶ palataanko tähän koskaan?

# Ketterä post covid?

- ▶ eräs ketterän perusolettamuksia on se, että tiimi jakaa yhteisen työtilan
  - ▶ palataanko tähän koskaan?



# Ketterä post covid?

- ▶ eräs ketterän perusolettamuksia on se, että tiimi jakaa yhteisen työtilan
  - ▶ palataanko tähän koskaan?



- ▶ miten ohjelmistokehitys onnistunut etääikana?

# Koe

- ▶ Koe tiistaina 19.12. klo 7-14 moodlessa (koeaika 4h)

# Koe

- ▶ Koe tiistaina 19.12. klo 7-14 moodlessa (koeaika 4h)
- ▶ Kaikki materiaali käytössä
  - ▶ copy paste tietysti kiellettyä
  - ▶ johtaa välittömään hylkäämiseen
  - ▶ entä ChatGPT?

# Koe

- ▶ Koe tiistaina 19.12. klo 7-14 moodlessa (koeaika 4h)
- ▶ Kaikki materiaali käytössä
  - ▶ copy paste tietysti kiellettyä
  - ▶ johtaa välittömään hylkäämiseen
  - ▶ entä ChatGPT?
- ▶ Koealueena kurssimateriaalin osat 1-5 sekä laskarit, paitsi
  - ▶ Git
  - ▶ poetry
  - ▶ GitHub Actions
  - ▶ unittest
  - ▶ Mock
  - ▶ Robot/Selenium

# Koe

- ▶ Koe tiistaina 19.12. klo 7-14 moodlessa (koeaika 4h)
- ▶ Kaikki materiaali käytössä
  - ▶ copy paste tietysti kiellettyä
  - ▶ johtaa välittömään hylkäämiseen
  - ▶ entä ChatGPT?
- ▶ Koealueena kurssimateriaalin osat 1-5 sekä laskarit, paitsi
  - ▶ Git
  - ▶ poetry
  - ▶ GitHub Actions
  - ▶ unittest
  - ▶ Mock
  - ▶ Robot/Selenium
- ▶ Vierailuluennot eivät kuulu koealueeseen

# Koe

- ▶ Koe tiistaina 19.12. klo 7-14 moodlessa (koeaika 4h)
- ▶ Kaikki materiaali käytössä
  - ▶ copy paste tietysti kiellettyä
  - ▶ johtaa välittömään hylkäämiseen
  - ▶ entä ChatGPT?
- ▶ Koealueena kurssimateriaalin osat 1-5 sekä laskarit, paitsi
  - ▶ Git
  - ▶ poetry
  - ▶ GitHub Actions
  - ▶ unittest
  - ▶ Mock
  - ▶ Robot/Selenium
- ▶ Vierailuluennot eivät kuulu koealueeseen
- ▶ Edellisten vuosien kokeita on nähtävillä vanhoilla kurssisivuilla
  - ▶ suureksi osaksi esseitä