

Friskrivningsklausul för Föreläsningen som ingenbett om Vänligen läs noga innan du tar del av denna presentation.

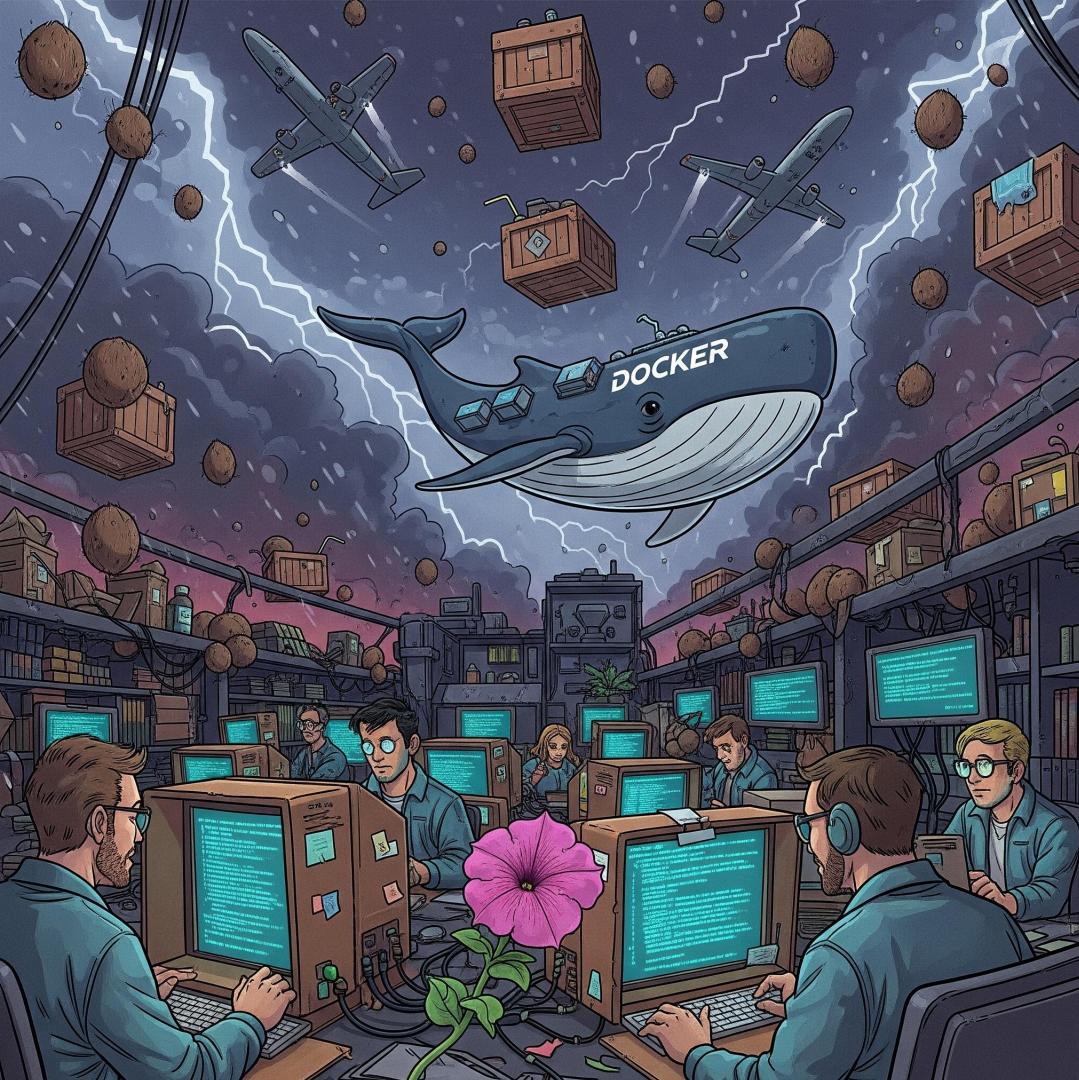
Din fortsatta närvaro och uppmärksamhet utgör ett fullt och oåterkalleligt godkännande av följande villkor.

Genom att stanna i detta rum accepterar du följande villkor, som är juridiskt bindande (om man inte bryr sig om lagar):

1. **Ingen Sanning är Helig:** Allt som sägs, visas eller antyds under denna presentation är föreläsarens personliga och djupt subjektiva syn på tillvaron. Vi garanterar inte att det är korrekt, logiskt eller ens begripligt. Du lyssnar på egen risk.
2. **Dina Känslor, Ditt Ansvar:** Om du känner dig förolämpad, upplyst eller bara plötsligt sugen på en kopp kaffe och en bulle – är det helt och hållt ditt eget ansvar. Föreläsaren tar inget juridiskt eller moraliskt ansvar för de reaktioner som kan uppstå. Men det finns bulle, som utlovat.
3. **Ord som Värmer:** Det kan förekomma lättare profaniteter. Detta är inte ett tecken på dålig uppfostran, utan snarare en konstnärlig förstärkning av det budskap jag vill förmedla. Om du är lättstött, blunda och tänkt på en fin regnbåge.

Tack för ditt mod. Må dina tankar vara lika fria som denna presentation.

Ove Lindström - Jag tar inget ansvar för någonting längre.



Cargo Cult Coding

*En historia som ingen sett om som
handlar om kokosnötter,
kod
och katastrofer.*

Historien bakom Cargo Cult



Begreppet Cargo Cult Science

Se
framgång

Kopiera
ritualerna

Förväntade
resultat

Misslyckas



FÖRELÄSNIGENS UPPLÄGG!

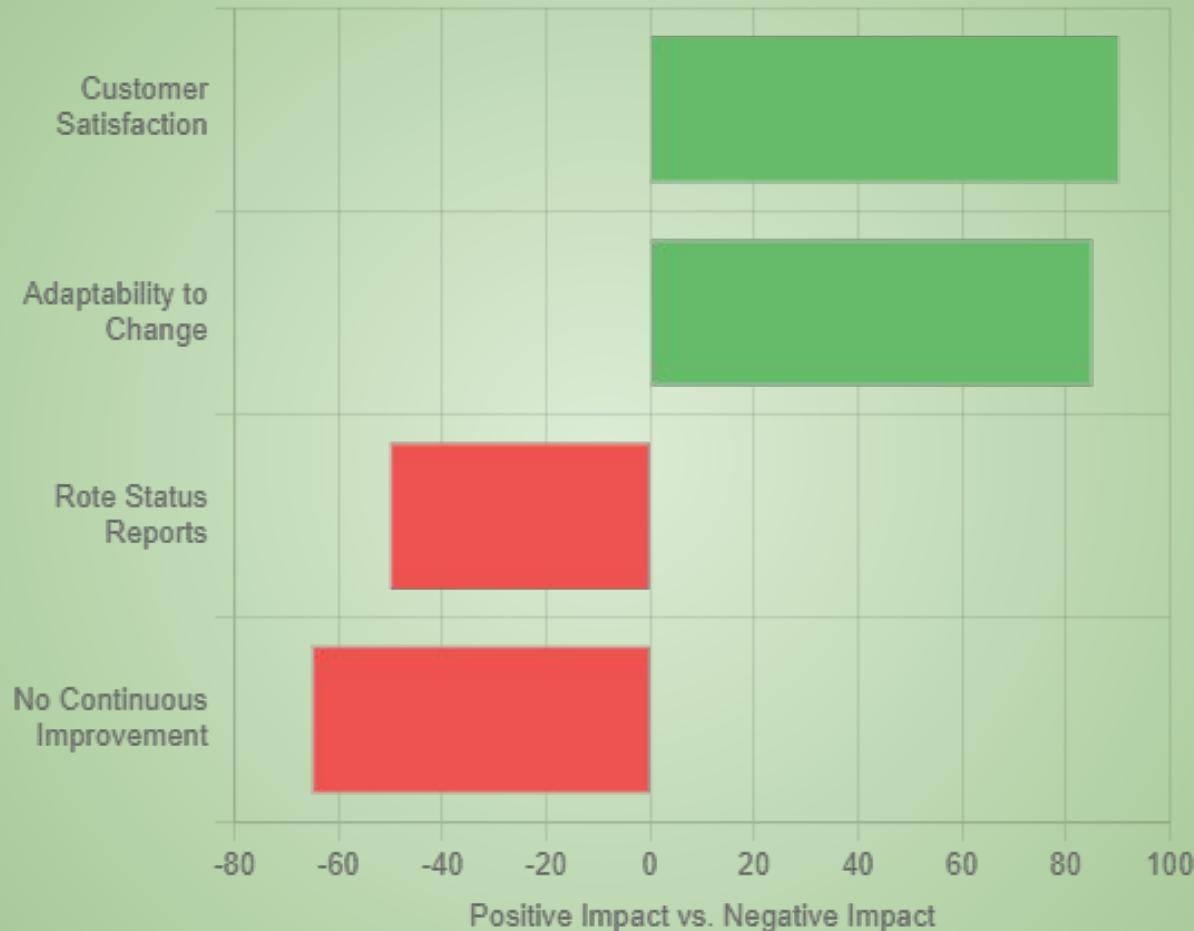


INSPIRÄT AV VERKLALITENT, FÖR ÄT NI NAT SK GÖRA SAMA FEL (KAKNDE!)

Agil utveckling

- Vad finns i lådan?
 - Kundnöjdhet
 - Hastighet
 - Anpassningsbarhet
 - Autonoma team
- Landningsbanorna:
 - Daily standups, Jira Boards, Spring Backlogs
 - Nya namn på gamla roller
 - Retrospectives
- Anti Pattern - Vad vi fick
 - Stand-ups som är statusrapporter
 - Ingen ständig förbättring
 - Samma rapportering retro efter retro
 - Vart tog automatiserade tester vägen?





Quick-fix:

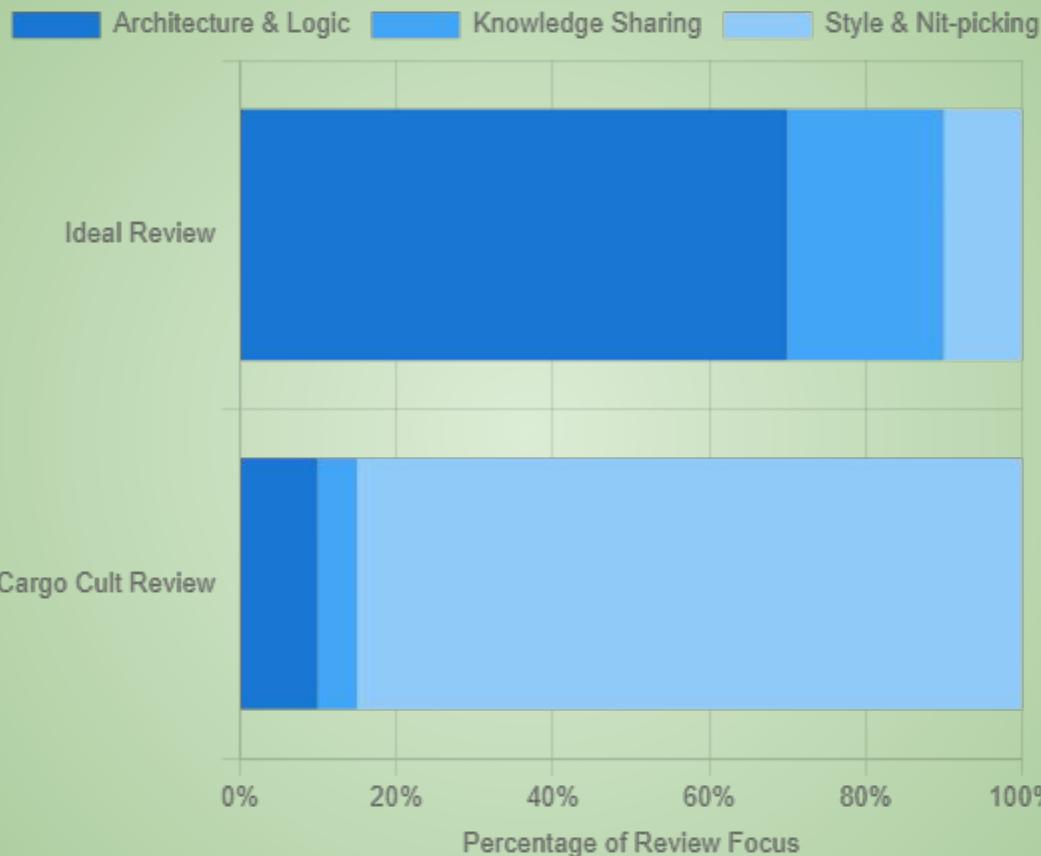
- Sluta fokusera på "vad gjorde du igår".
- Fråga istället "vilket värde skapades".
- Prioritera det viktigaste.
- Levererat värde, inte aktiviteter.



Code Reviews

- Vad finns i lådan?
 - Snabbare time to market
 - Bättre säkerhet
 - Pålitlighet
 - Mer delad kunskap (mellan de autonoma teamen)
- Landningsbanorna:
 - Obligatoriska Pull Request
 - SAST, DAST och Linters
 - ISO27001 och SOC2
- Anti Pattern - Vad vi fick
 - Oändliga reviews - "död genom tusen pull requests"
 - Målarbokssäkerhet
 - Skolpjäser för SOC2-auditors





94% av de företag som kommit igenom en ISO27001 och/eller SOC2-audit känner sig säkra efter granskningen.

62% av alla företag som blir hackade eller får en säkerhetslåcka är ISO27001 och SOC2-certifierade.

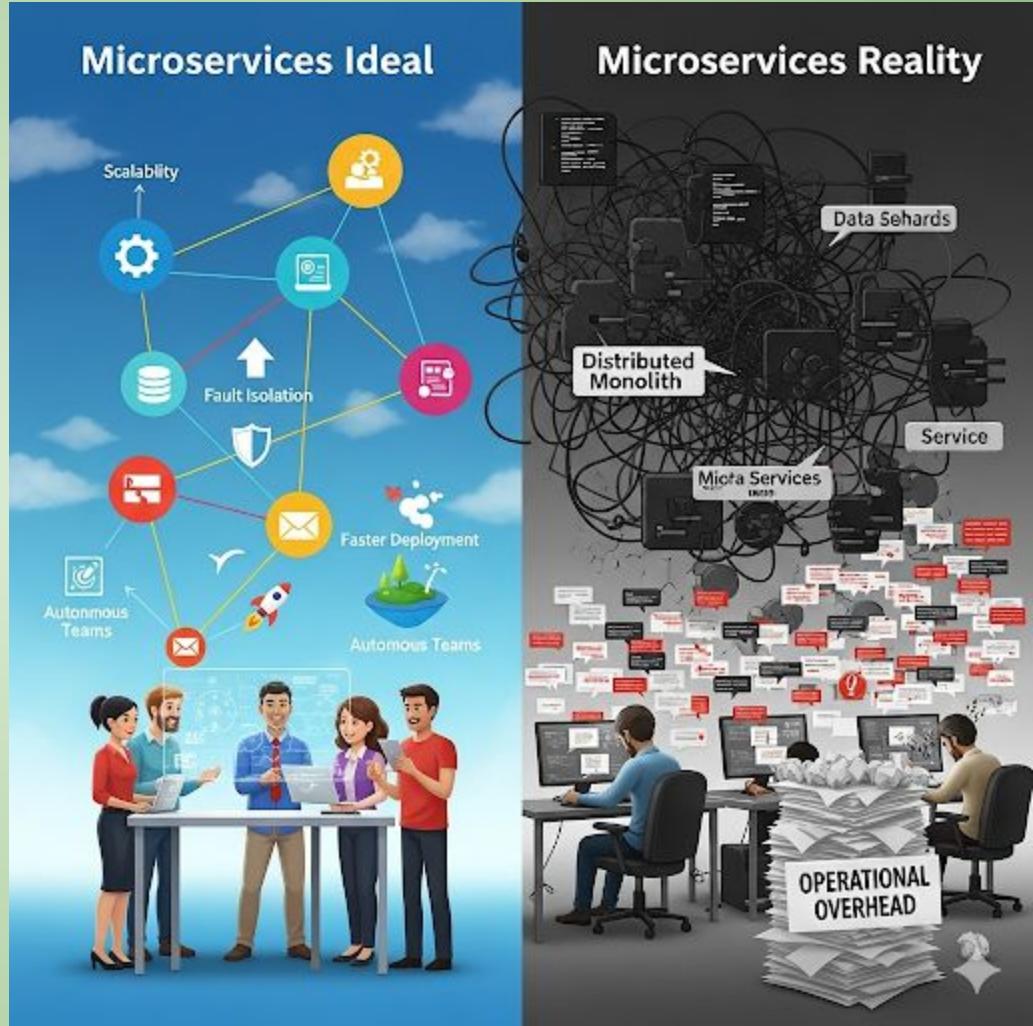
Quick-fix:

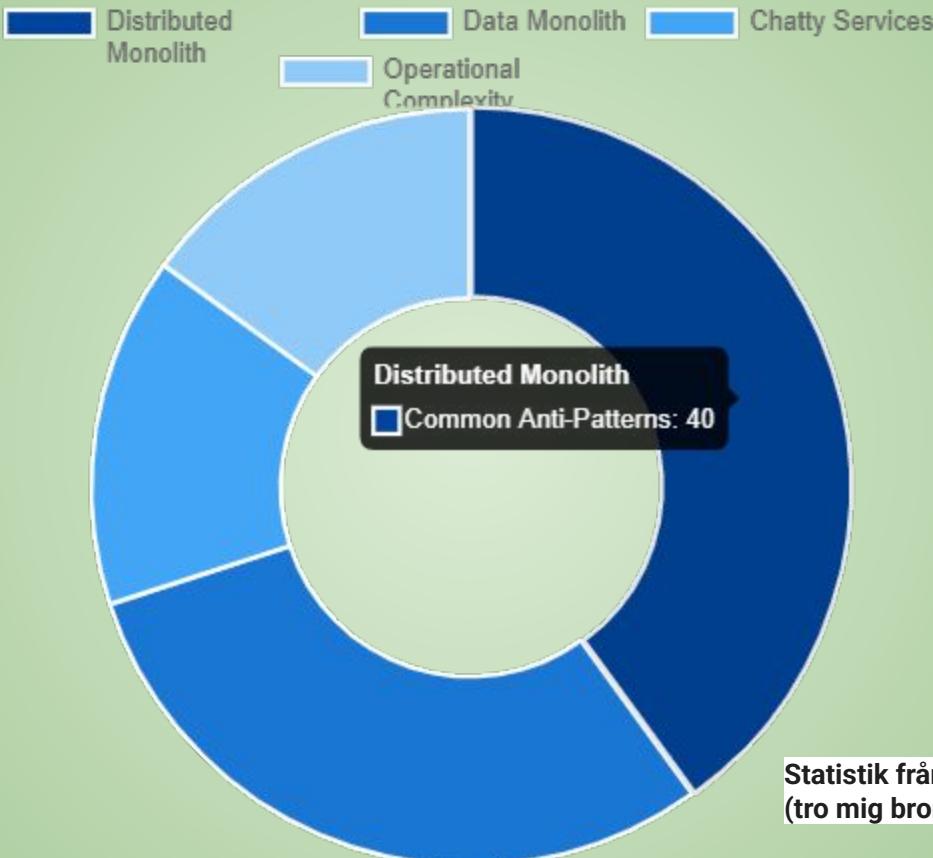
- Fokusera på logiken och säkerheten.
- Automatisera styling vid push
- Låt varje utvecklare ha sin stil när de kodar.



Microservices

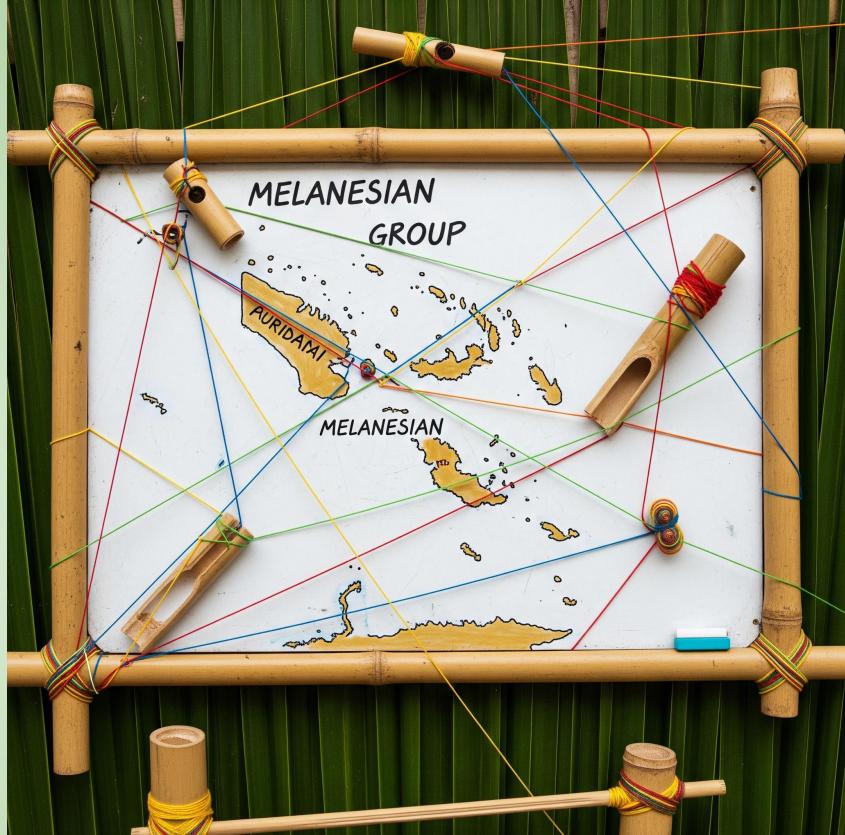
- Vad finns i lådan?
 - Skalbarhet!
 - Fel-isolering
 - Snabbare driftsättning
 - Autonoma team (favorit i repris)
- Landningsbanorna:
 - API Gateways
 - Orketstreringsramverk
 - Upplösta tjänster
- Anti Pattern - Vad vi fick
 - Distribuerade monoliter
 - Death by 1000 data shards
 - Pladdriga tjänster som använder all bandbredd
 - Operationell överhettnig





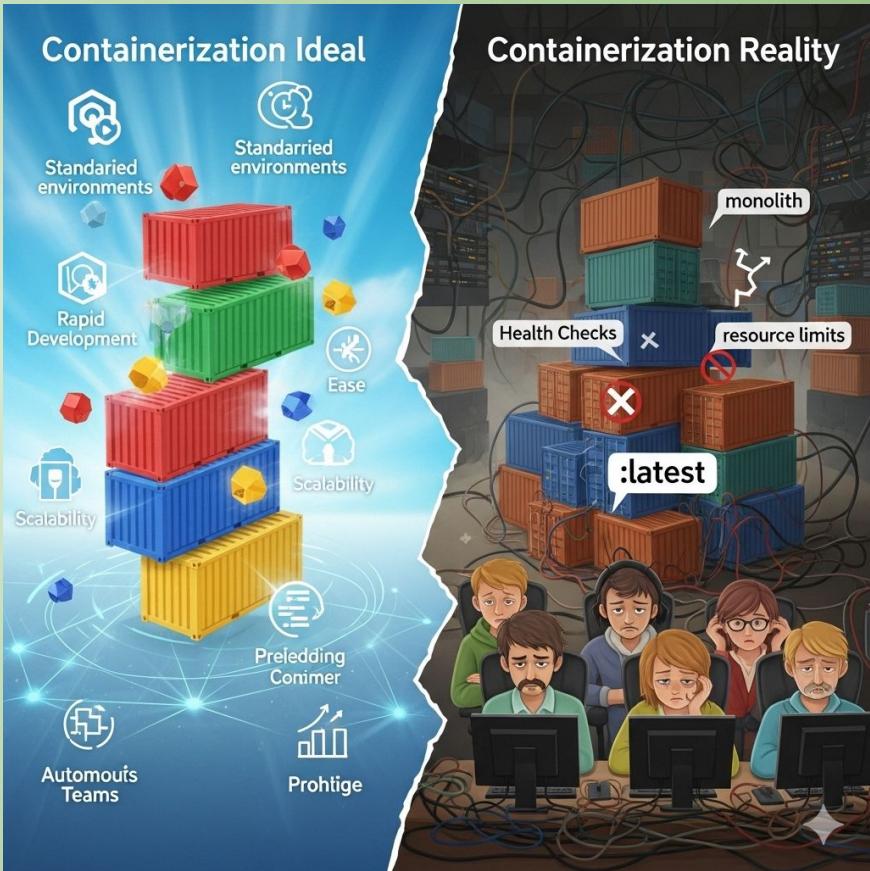
Quick-fix:

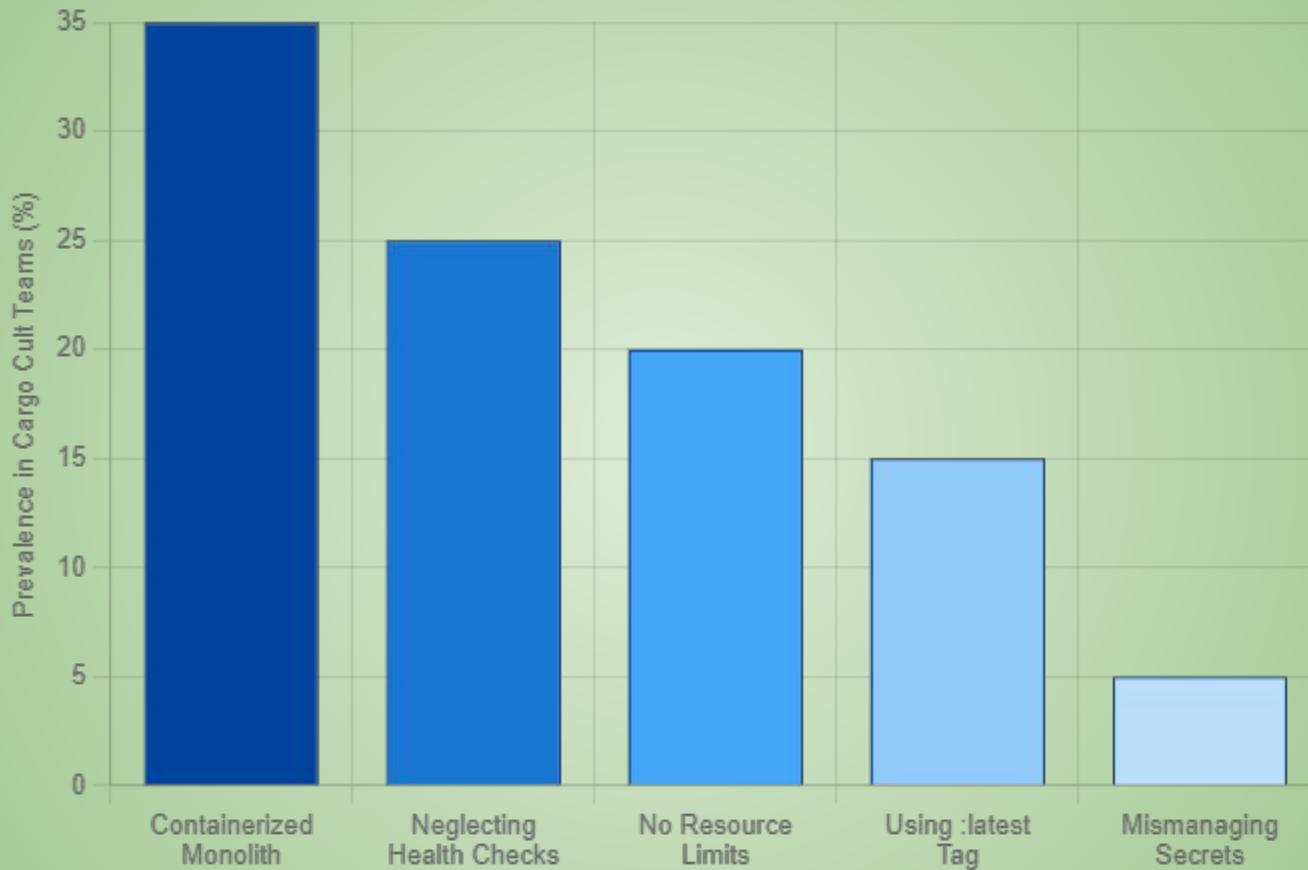
- Förstå verkligen hur de olika gränserna mellan dina tjänster ser ut.
- Om du inte kan rita det på en whiteboard är det förmodligen svårt att utveckla.



Containerisering och Orkestrering

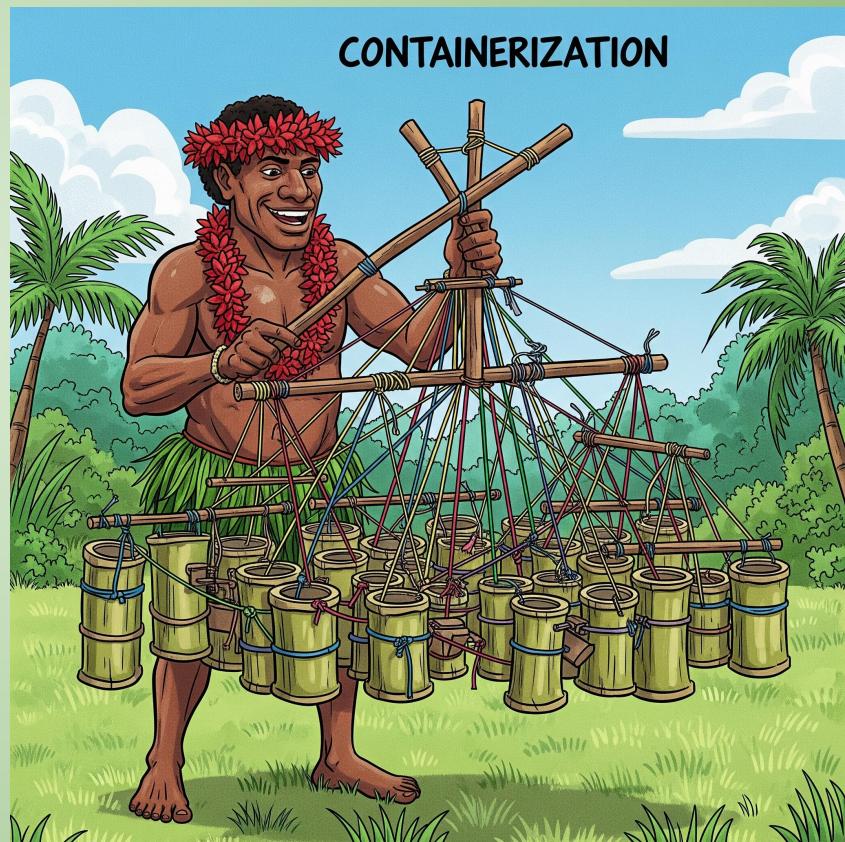
- Vad finns i lådan?
 - Standardiserade miljöer
 - Snabb utveckling
 - Skalbarhet
 - Autonoma team (igen?)
- Landningsbanorna:
 - Docker containers
 - Kubernetes
 - YAML files
 - kubectl, kubectl, kubectl, kubectl
- Anti Pattern - Vad vi fick
 - Containeriserade monoliter
 - Ingen hälsokontroll
 - Vadå resurs-gränser?
 - `:latest` är enklast, då slipper man uppdatera och köra `kubectl` hela tiden





Quick-fix:

- Man kommer långt med Docker Compose.
- Börja med “serverless” innan du går till Kubernetes.
- Förstå behoven innan implementering



DevSecOps

- Vad finns i lådan?
 - Snabbare time to market
 - Kunskapsdelning
 - Reducerad risk
 - Autonoma team (men ge dig!)
- Landningsbanorna:
 - Automatiserade byggen
 - SAST/DAST/SCA scanners
 - "shift-left" QA
- Anti Pattern - Vad vi fick
 - Automatisering av trasiga processer
 - Verktyg över kultur
 - Säkerhet som en automatisk checkbox
 - shift-leftier development

MFA REQUIRED DUE TO
SECURITY POLICY 14.5 AND 18.1

Ideal Flow

Culture of Collaboration



Refined Process



Enabling Tools

Cargo Cult Flow

Buy Tools First



Automate Broken Process



Scale Inefficiency

Quick-fix:

- Ha en Top 3 lista över verkliga hot som kan uppstå i ert system.
 - Fokusera på de 3 så följer resten efter automatiskt.
 - Om systemets gränser ändrar sig, uppdatera Top 3.

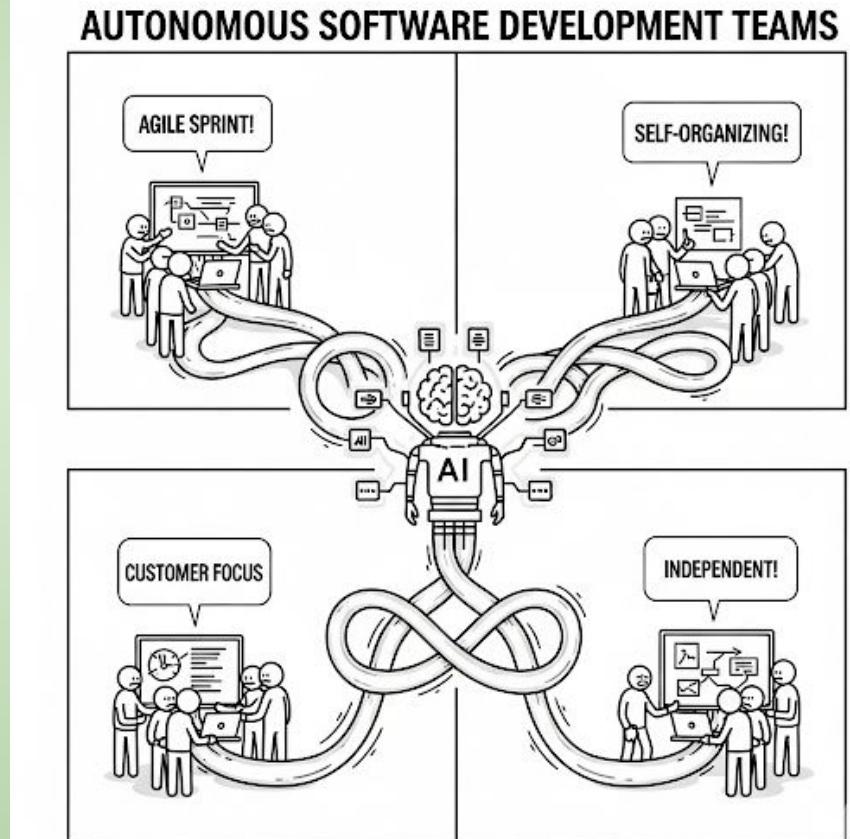


Promised Benefits (Crates) Actual Outcomes (Symptoms)



Att bryta ett mönster - Sluta apa efter, börja förstå

- Börja med “Varför” 🧠
- Investera i Kulturen 🤝
- Omfamna lärandet 📈
- Fokusera på människorna 🧑





Tack!