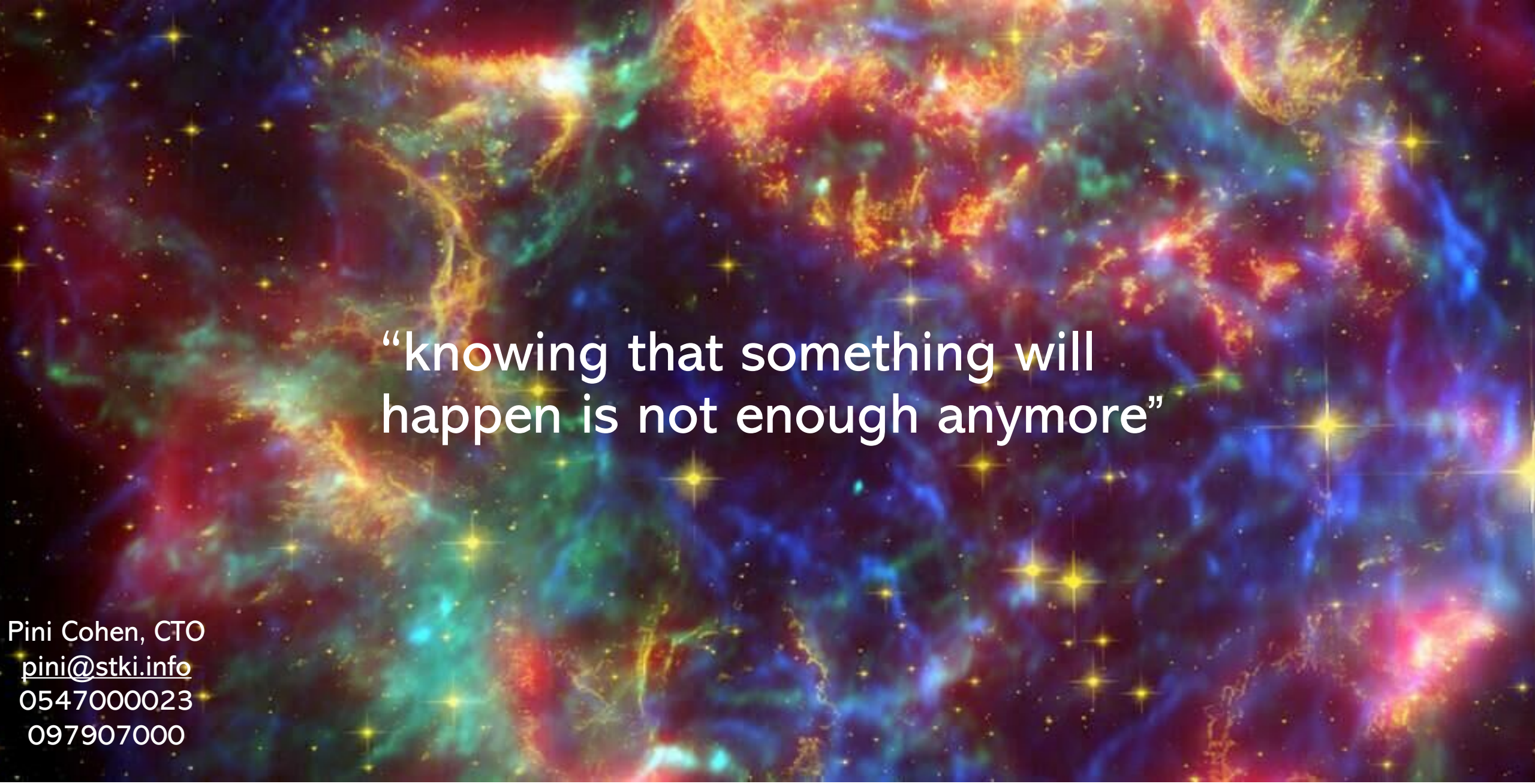


Pini Cohen, CTO
pini@stki.info
0547000023
097907000

Agenda

- COVID-19 IT market implications
- Infra. & Ops. trends, organization, SRE
- Dev. & architecture trends, Self Contained Systems (SCS), EDA (Event Driven), microservices sprawl
- Cloud vs. IT

Trends presentation

- The good
- The bad
- The evil

STKI IT Market Forecast:

FORECAST ONE

Assumes that the main part of the restrictions intended to prevent the further spread of the virus will be lifted gradually by the end of June, so that in the second half of the year the economy will return to activity without limitations on movement or employment, and will switch to a path of growth.

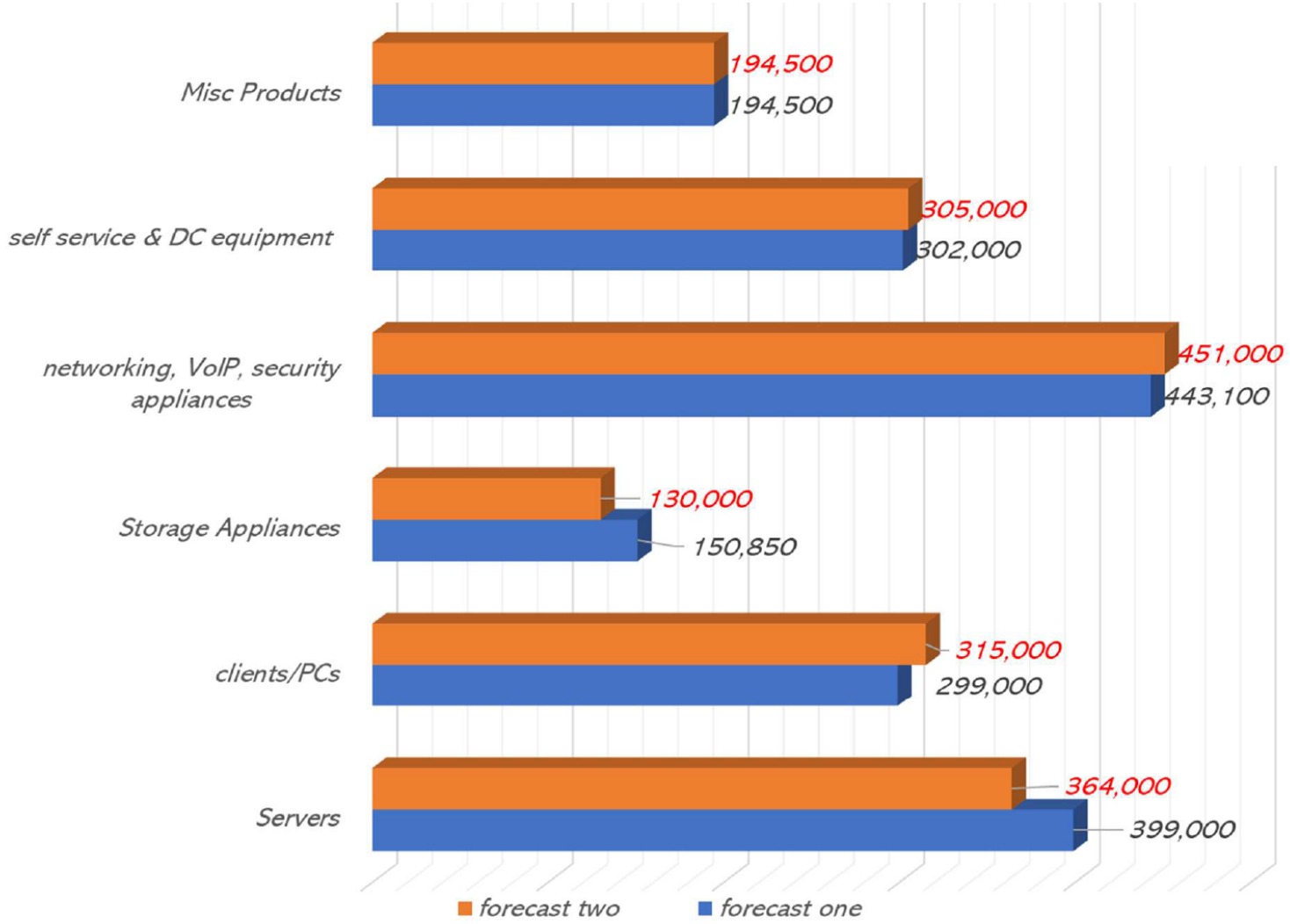
Some of the lessons the crisis taught us will NOT be implemented in the immediate future and will be postponed. A lot of activity will be similar but smaller than in previous years.

FORECAST TWO

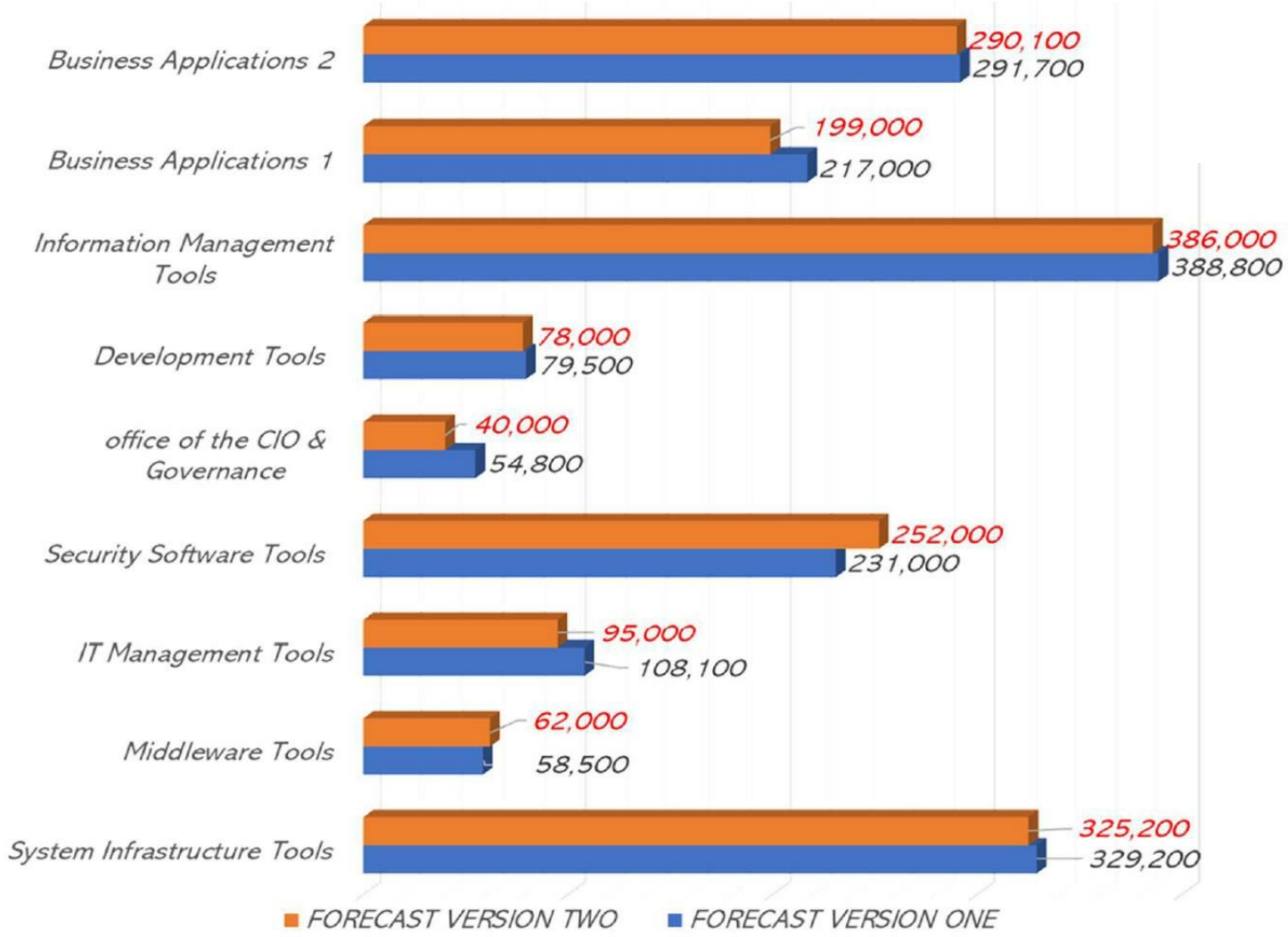
Assumes that the steps taken will be temporary, but with year long uncertainty . Some restrictions will be lifted but some will be implemented again. All based on the specifics of the contagion and the load placed on hospital infrastructures in Israel.

Lessons from the crisis for example:
application usability, user friendly ecommerce, better internet connections, work from home and many others will be implemented and most old projects will be stopped or changed.

STKI Market study: HW

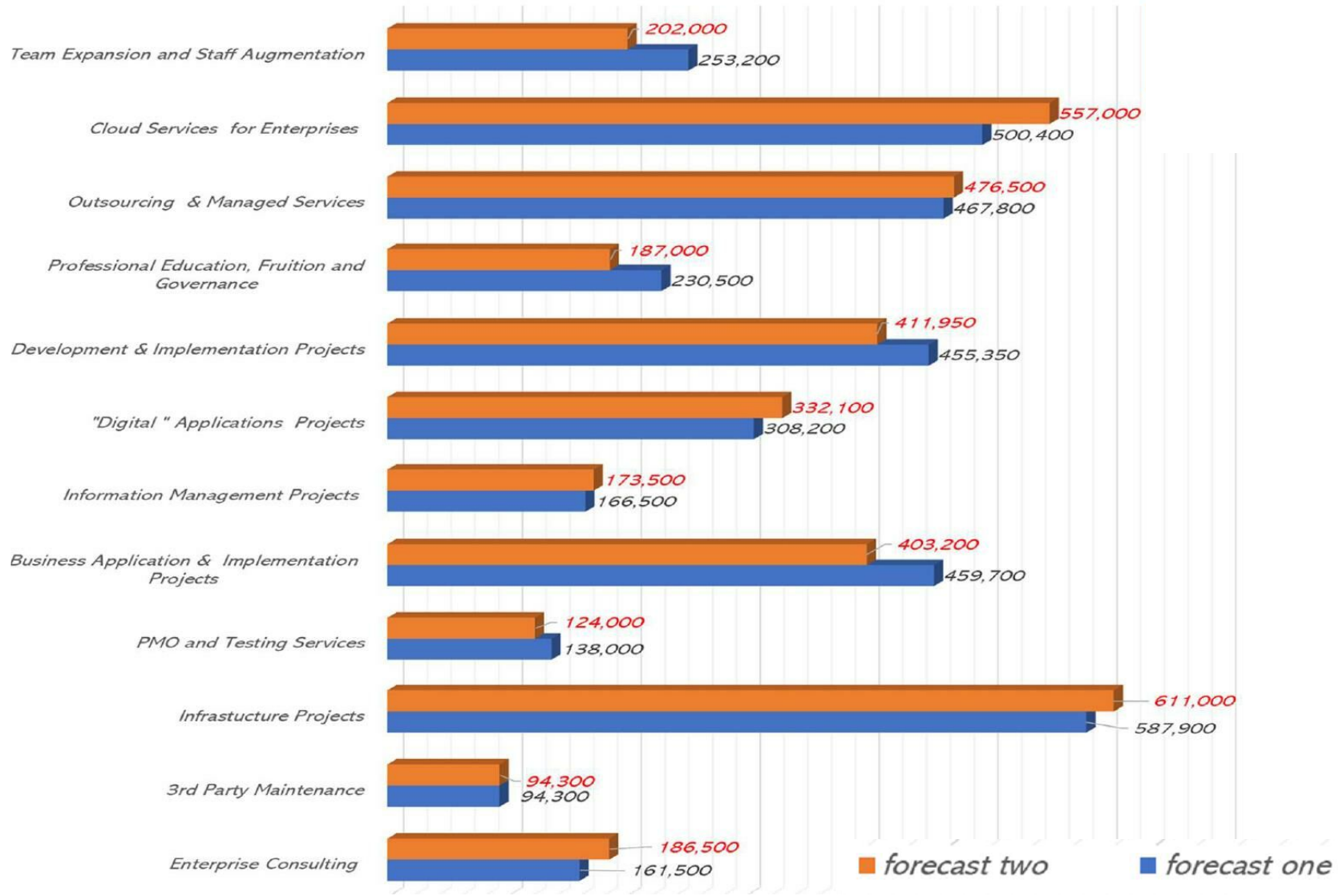


STKI Market Study: SW



stki.info

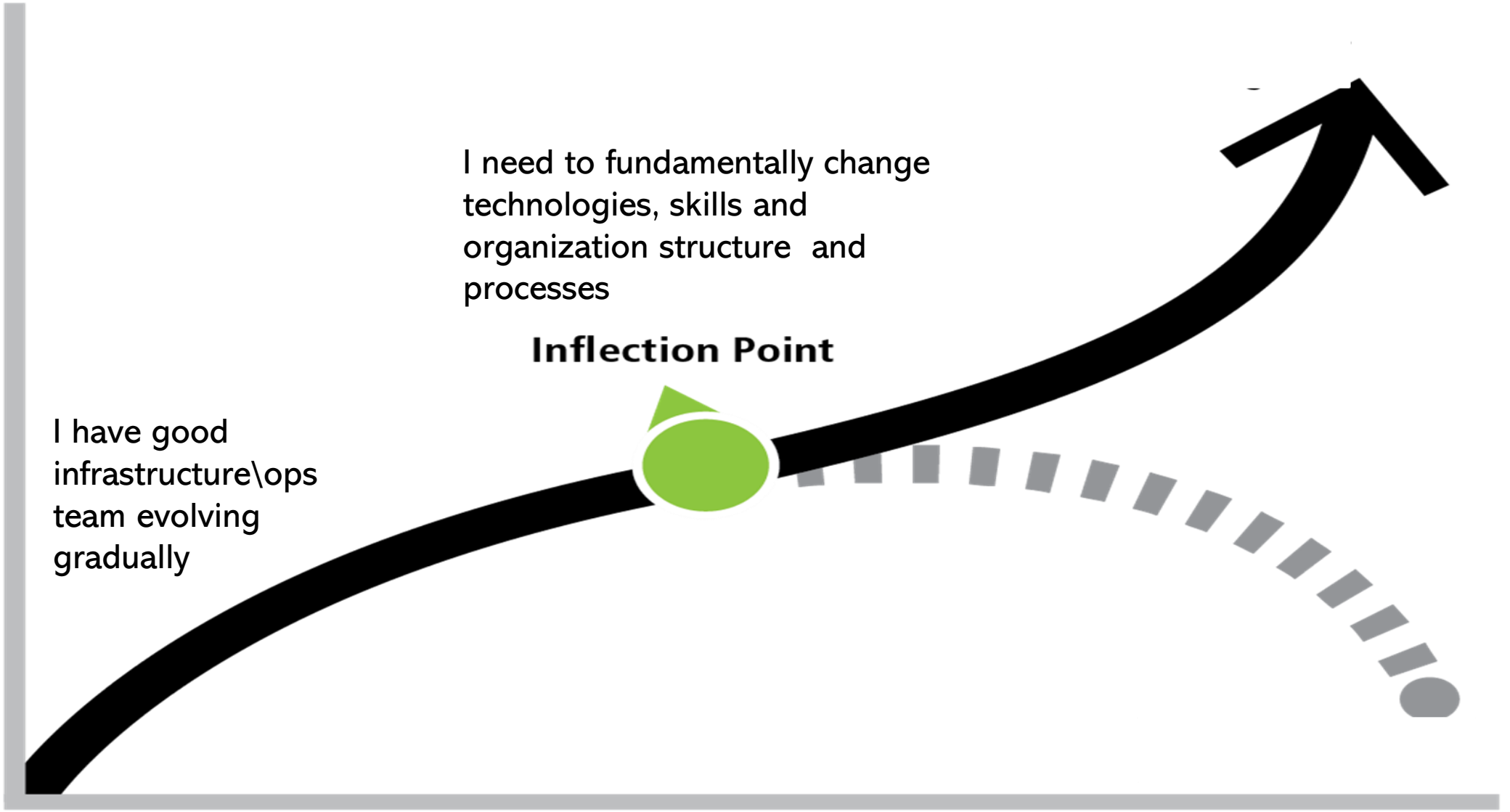
STKI Market Studv: Services



Agenda

- COVID-19 IT market implications
- Infra. & Ops. trends, organization, SRE
- Dev. & architecture trends, Self Contained Systems (SCS), EDA (Event Driven), microservices sprawl
- Cloud vs. IT

Infra\Ops inflection point



The evolution of infrastructure \ operations

Provision storage

Infra	• Takes week (RAID groups, striping, rebuilds)
Infra	• Takes a minute (array improvements)
Infra	• Automated so infra can scale

Dev (manually)	• Self Service portal
	  • Just storage
	  • “Bigger” (combined) service

DevOps scripts	• Service catalog API’s
DevOps & SRE	• “Cloud native needs”: CI\CD , Blue\Green, Fast recovery

The evolution of infrastructure \ operations

- Different tasks
- Different scale
- Different processes
- Different technologies (cloud, containers)
- Different skill

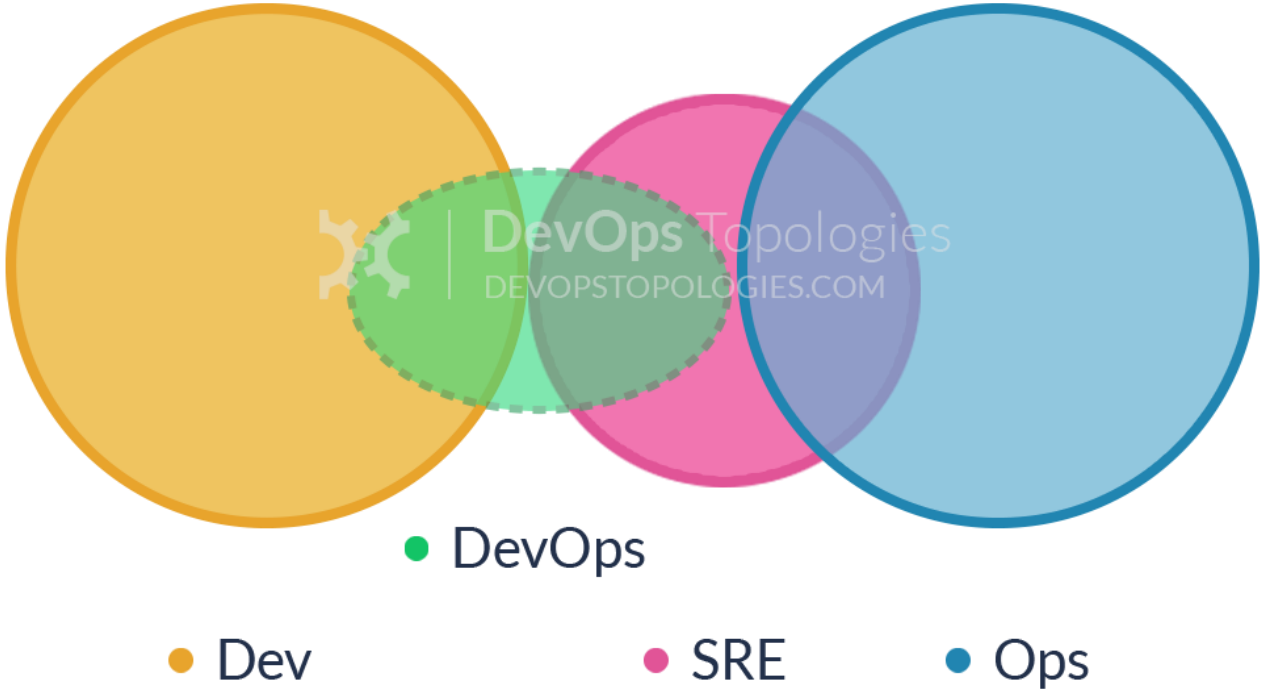
Head of DBA team:

CEO of boutique networking integrator:

CTO of governmental IT:

From Ops to Devops to SRE (Site Reliability Engineer)

- Discipline originated in Google that incorporates aspects of software engineering and applies them to infrastructure and operations problems.
- The main goals are to create scalable and highly reliable software systems.



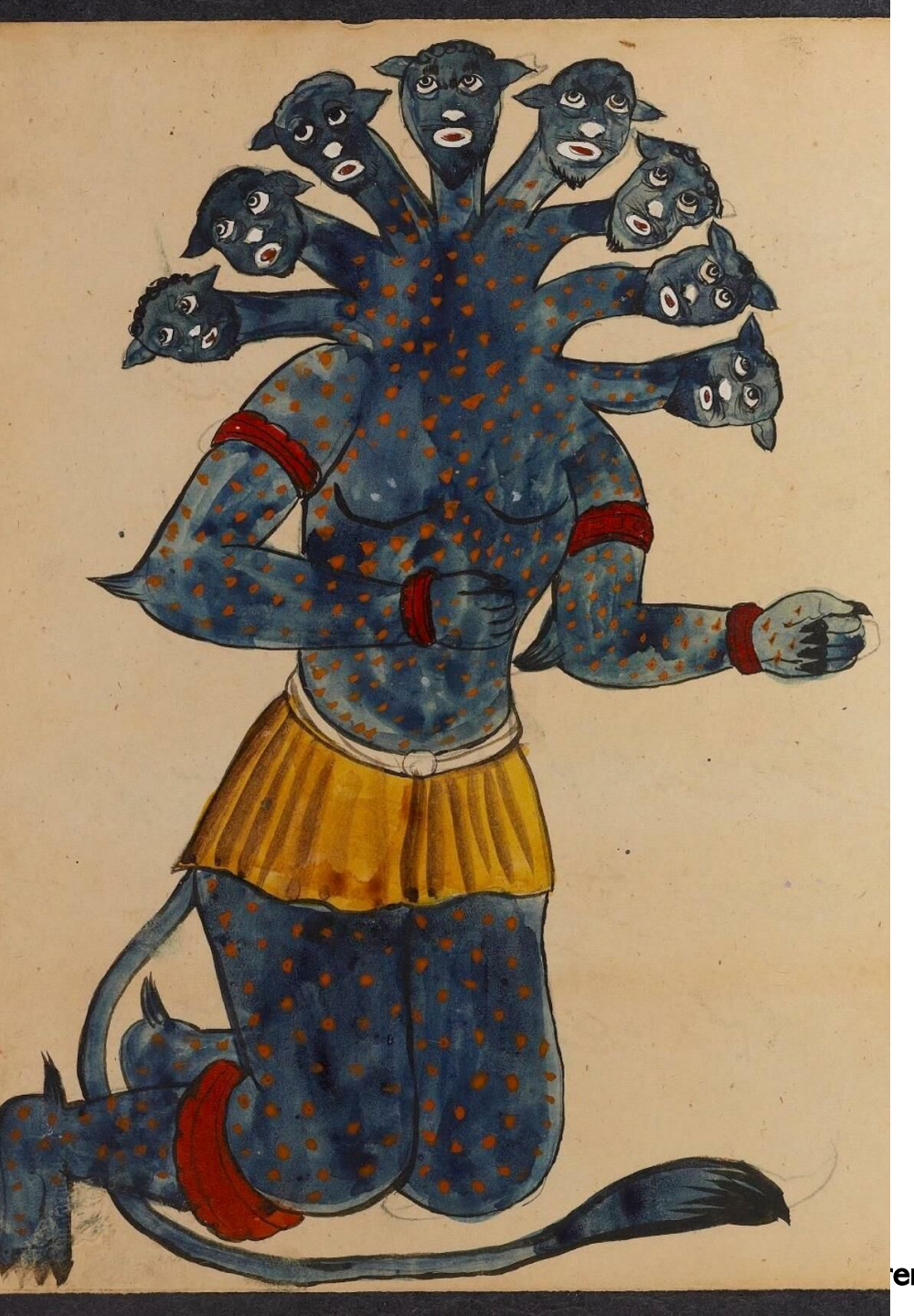
Source: https://techbeacon.com/enterprise-it/5-ways-site-reliability-engineering-transforms-it-ops

Example of infra\ops changing tasks:

- From MTBF (Mean Time Between Failure) to MTTR (Mean Time To Repair)
  - MTBF implies less changes, rigid (yet strong) infrastructure
  - MTTR implies automation, fast actions when failure happens, rebuild instead of repair (immutable components)

“What exactly is Site Reliability Engineering, as it has come to be defined at Google? My explanation is simple: SRE is what happens when you ask a software engineer to design an operations team.”

- Benjamin Treynor Sloss,
Vice President, Engineering, Google



SRE principles

- Accept failure as normal
- SRE quantifies failure and availability in a prescriptive
- SRE mandates blameless post mortems
- SRE encourages developers and product owners to move quickly by reducing the cost of failure
- Leverage tooling and automation
- Measure everything
- SRE fundamentally believes that systems operation is a software problem

emove source or attribution from any slide, graph or portion of graph

Source: wikipedia

Google SLI, SLO, SLA

Service Level Indicators (SLIs)

- A carefully defined quantitative measure of some aspect of the level of service that is provided
- **request latency** / **error rate** (often expressed as % of all requests received) / **system throughput**,

Service Level Objectives (SLOs)

- Lower bound ≤ SLI ≤ upper bound
- Define the lowest level of reliability, and state that as your Service Level Objective (SLO).

Service Level Agreements (SLAs)

- SLA is a looser objective than the SLO. Alternatively the SLA might only specify a subset of SLO metrics.
- I.e. availability SLA of 99.9% over 1 month with internal availability SLO of 99.95%
- A promise to someone using a service that its availability should meet a certain level over a certain period, and if it fails to do so then some kind of penalty will be paid (partial refund of subscription fee paid by customers for that period, or subscription time added for free)

Full story: https://cloudplatform.googleblog.com/2017/01/availability-part-deux--CRE-life-lessons.html

SRE quantification

SRE vs. Devops

SRE

- Operations
- Incident response
- Post Mortems
- Monitoring, Events, Alertings
- Capacity planning
- Primary focus: Reliability

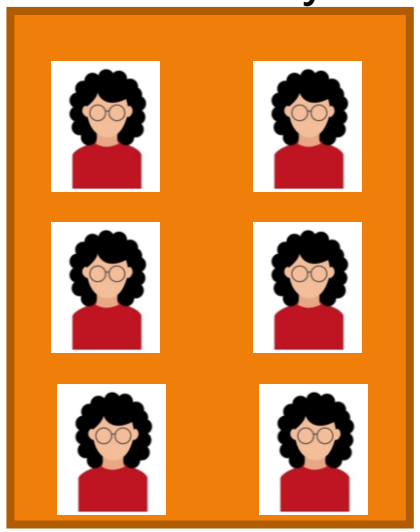
Devops

- Delivery
- Release automation
- Environment builds
- Config management
- Infrastructure as code
- Primary focus: Delivery Speed

Source: Atlassian

Organization structure: Technology centered

Technology silo: gives services centrally



Business centered 1

Business centered 2

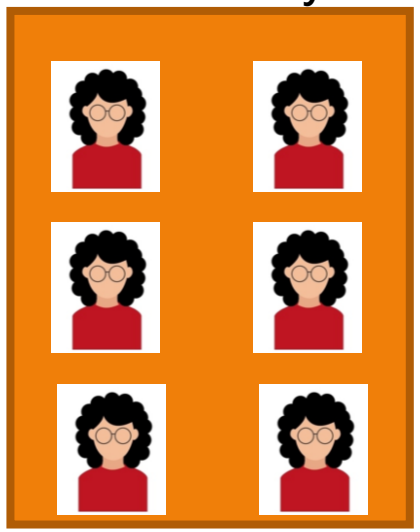
Business centered 3

Keeps technology standards
Better operations
Most efficient with people and technology spend (from silo point of view)
Kills the business

Business (Application) is kicking back

Which is worst? Technology centered vs. Business centered

Technology silo: gives services centrally



Business centered 1

Business centered 2

Business centered 3

Keeps technology standards
Better operations
Most efficient with people and technology spend (from silo point of view)
Kills the business

Technology is within business

CTO tries to give guidance

Business centered 1

Business centered 2

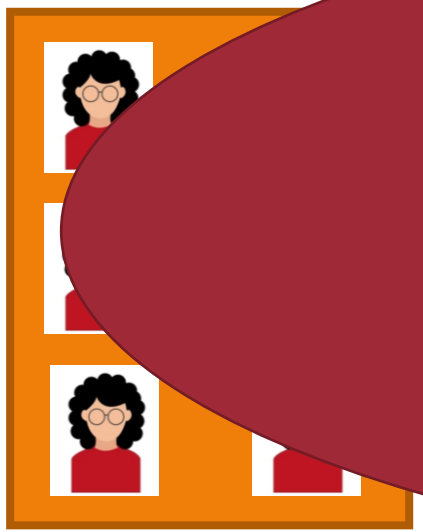
Business centered 3

Business is happy until comes the need to integrate several businesses (impossible)
Poor standards
Not efficient (skills, technology usage)
Harder to operate as a whole

stki.info

Which is worst? Technology centered vs. Business centered

Technology silo: gives services centrally



Keeps technology standards
Better operations
Most efficient with people and technology spend (from silo point of view)
Kills the business

Technology is

Business centered 1

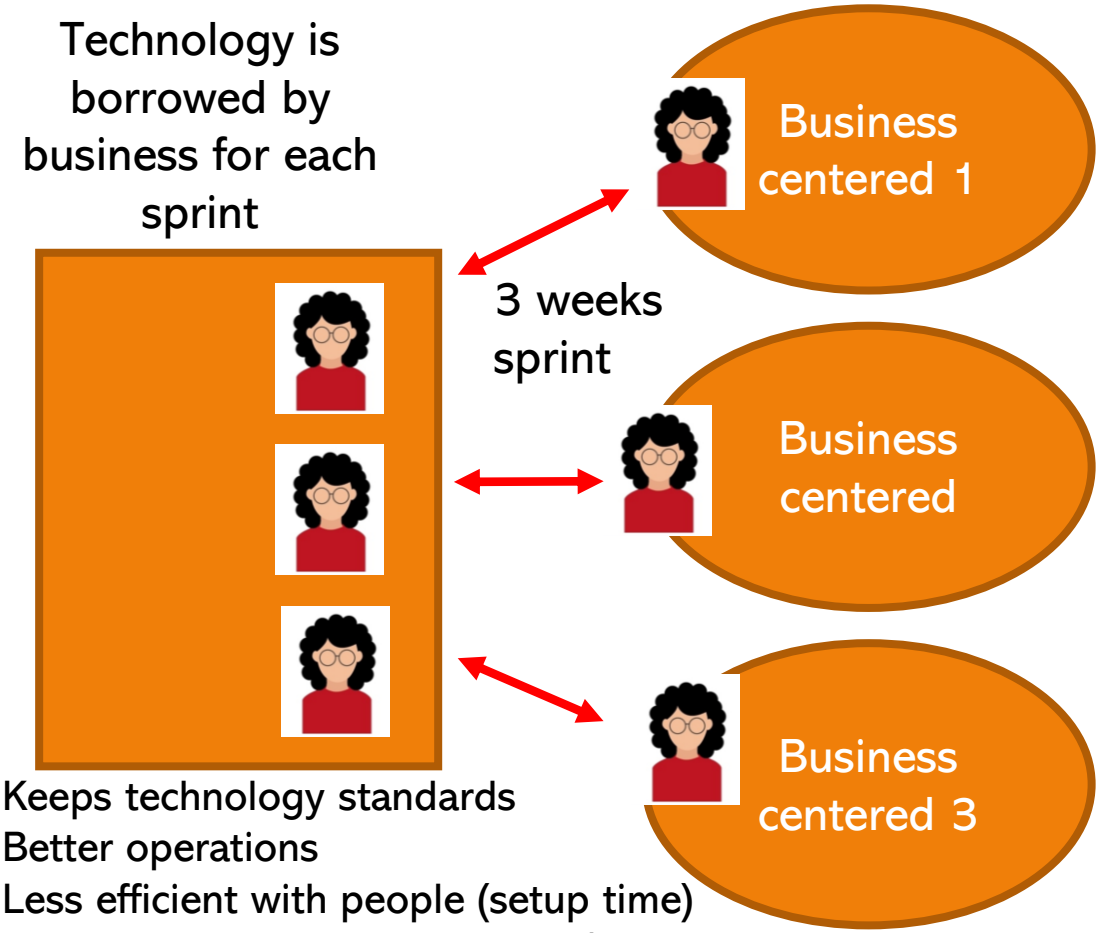
Business centered 3

happy until comes the need to integrate several businesses (impossible)
Poor standards
Not efficient (skills, technology usage)
Harder to operate as a whole

And the answer is: technology silos is the worst!!

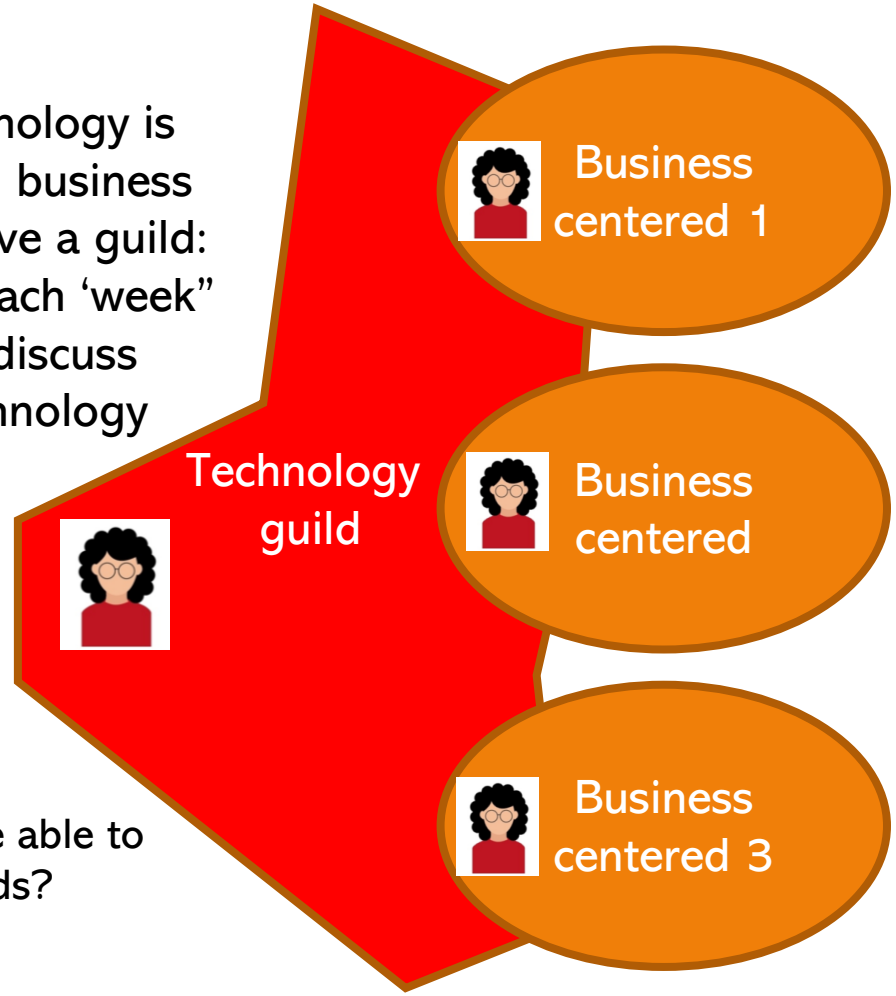
Circle the rectangle?

Technology is borrowed by business for each sprint



Keeps technology standards
Better operations
Less efficient with people (setup time)
Business is happy but not 100%

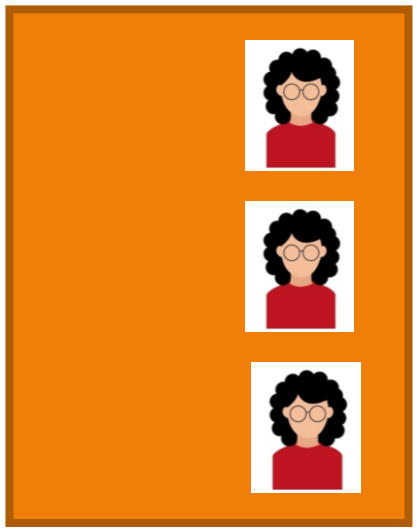
Technology is within business but have a guild: meet each 'week" to discuss technology

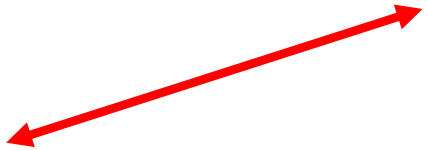


Will the guild be able to enforce standards?

Circle the rectangle?

Technology is delivered by
“self service” and by spreading
some of the skills





Self service &
central
guidance

Keeps technology standards
Better operations
Less efficient (need to do some reskill)

Agenda

- COVID-19 IT market implications
- Infra. & Ops. trends, organization, SRE
- Dev. & architecture trends, Self Contained Systems (SCS), EDA (Event Driven), microservices sprawl
- Cloud vs. IT

IDC: “Hyperagile apps”

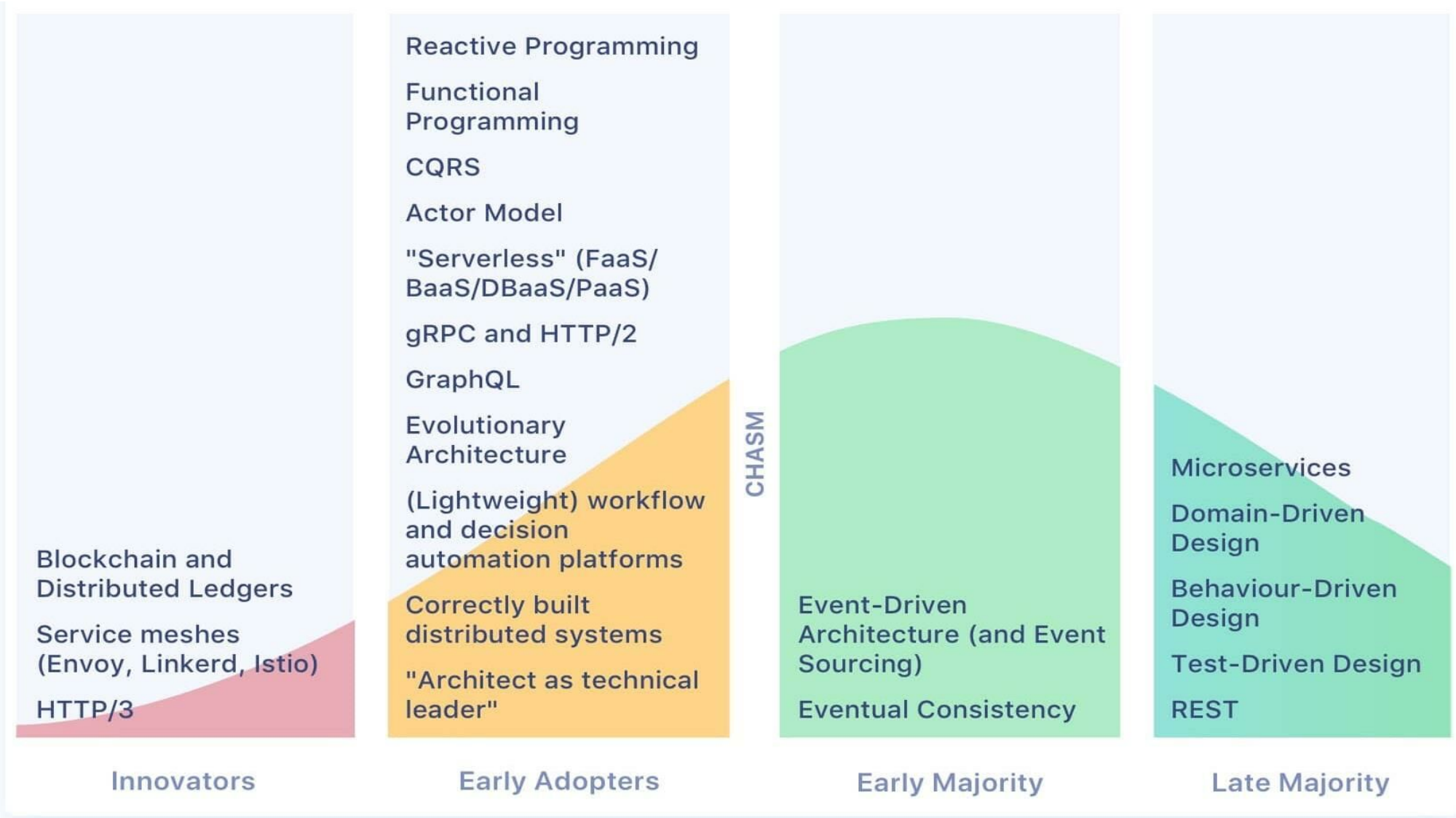
- By 2022, 90 percent of all new apps will feature microservices architectures that improve the ability to design, debug, update, and leverage third-party code.
- Hhyperagile apps are highly:
  - Modular
  - Distributed
  - Continuously updated,
  - Leveraging cloud-native technologies such as containers and serverless computing.

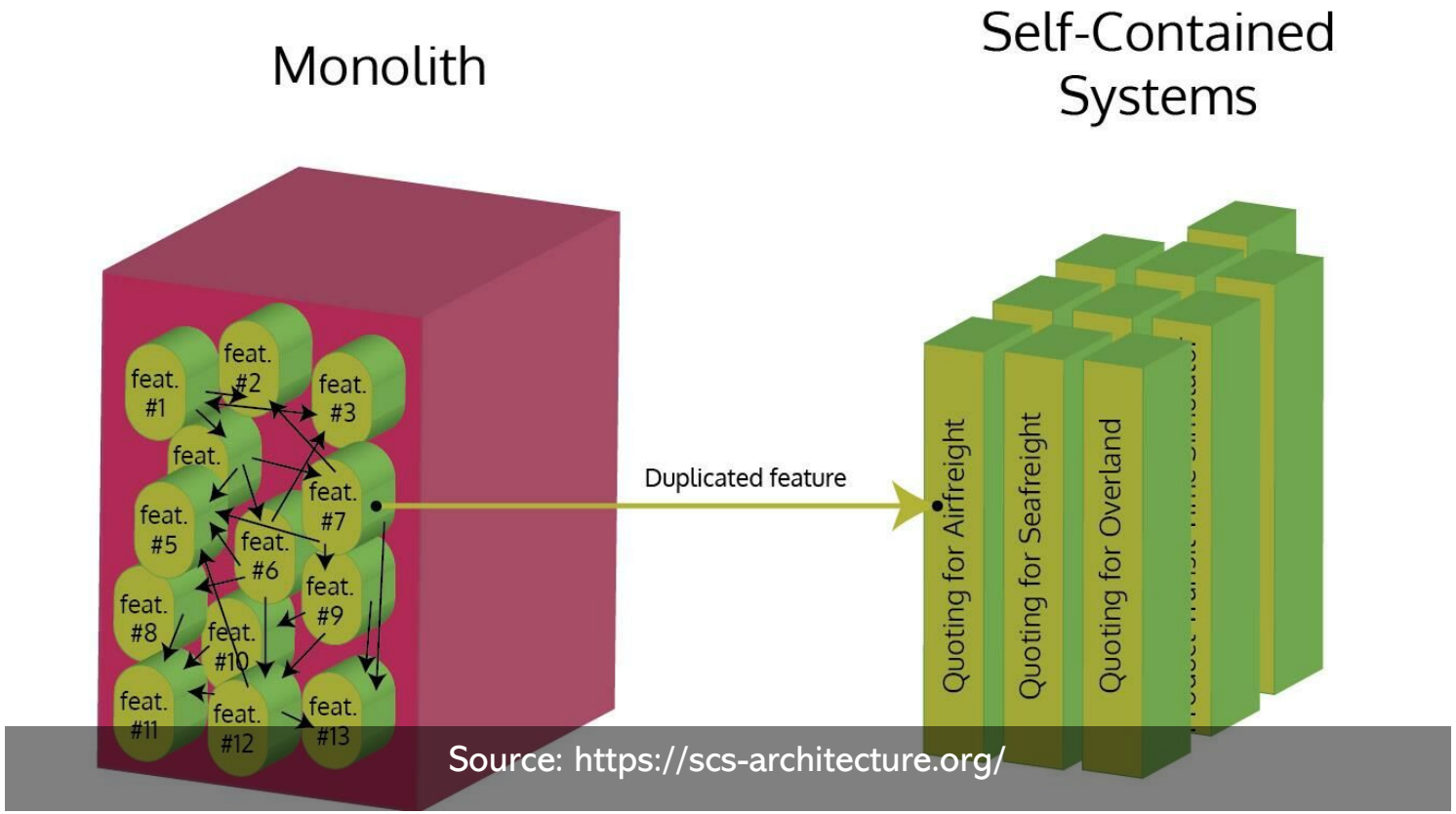
Source: IDC

Tomorrow is now:

Software Development
Architecture and Design 2019 Q1 Graph
http://infoq.link/architecture-trends-2019

InfoQ

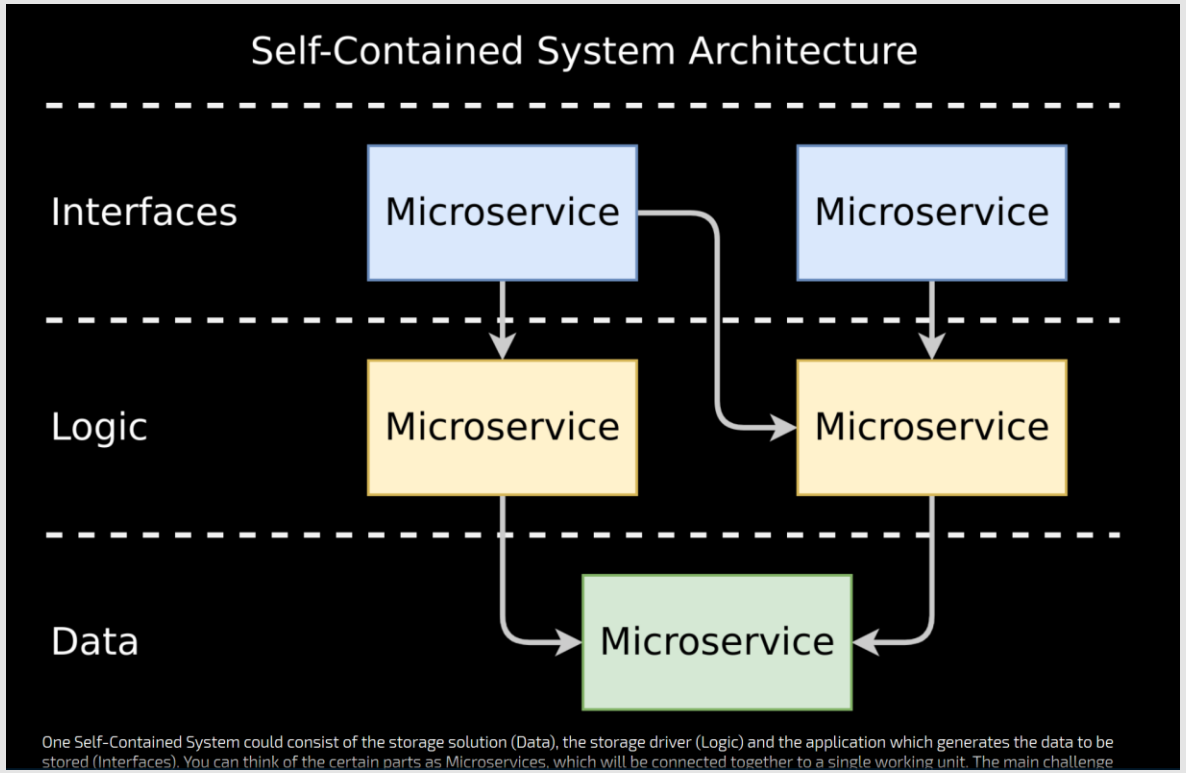




Self-contained system (SCS):

- Each SCS is an autonomous (web) application
- For the SCS's domain, all data, the logic to process that data and all code to render the web interface is contained within the SCS.
- An SCS can fulfill its primary use cases on its own, without having to rely on other systems being available.
- Communication with other SCSs or 3rd party systems is asynchronous wherever possible via well defined API's

SCS vs. Microservices

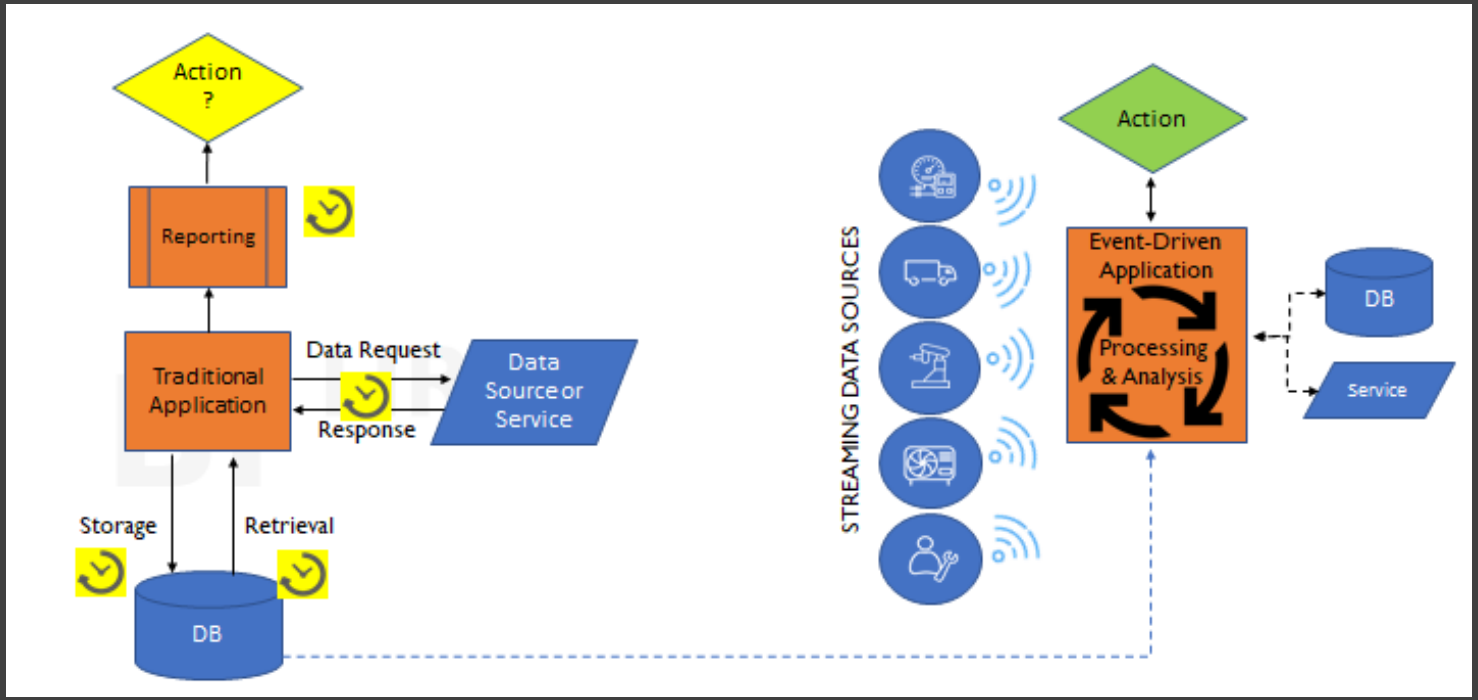


- A microservice is probably smaller than an SCS
- There are usually fewer SCSs than microservices. E-commerce shop might have 5 to 25 SCSs i.e. for billing, order processing and 100s of microservices.
- SCSs should ideally not communicate with each other while this is fine for microservices. If SCSs communicate – a-synchronous is preferred
- SCSs have a UI, while microservices might separate the UI from the logic in its own service

Source: http://www.quaperna.com/de/nachrichten-und-publikationen/tech-blog/214-practial-guide-to-self-contained-systems

Source: https://scs-architecture.org/

Event driven architecture - EDA



- With an **event-driven system**, the capture, communication, processing, and persistence of **events** are the core structure of the solution
- An **event** is any significant occurrence or change in state for **system** hardware or software.

Event driven - "If you love someone set him free"

- Traditional programming:

Make_Order {

…

…

Call Order_Fulfilment (id of order).

Wait for response (ack)

- }

- EDA programming:

Make_Order {

…

…

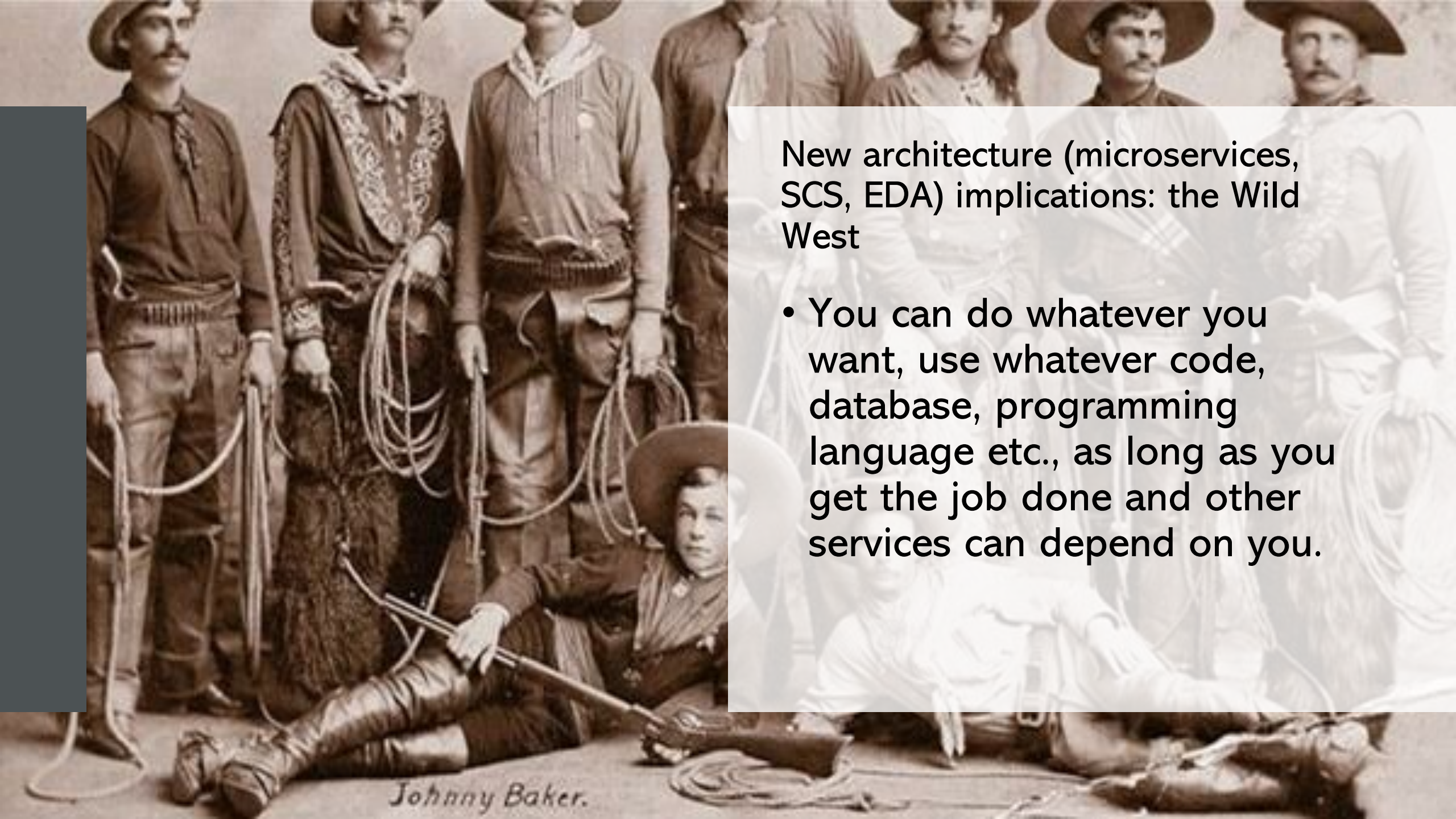
Publish event: Order_created(id of order)

//do not wait

- }

Event driven benefits & drawbacks

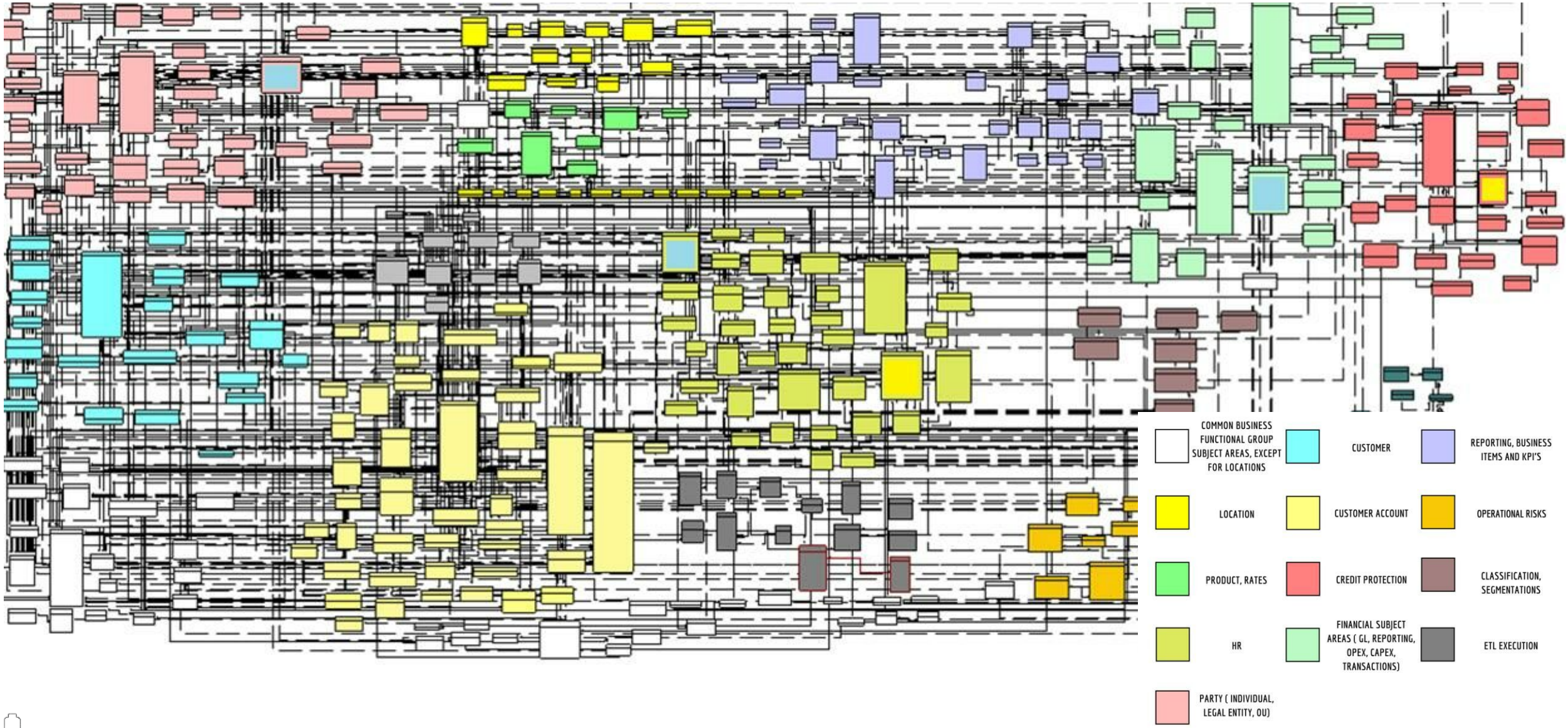
- Enables agile business processes
- Enables work of separate teams
- Fits microservices, self contained systems, devops, serverless
- Distributed transactions are difficult!!
- Needs to reskill architects & programmers



New architecture (microservices, SCS, EDA) implications: the Wild West

- You can do whatever you want, use whatever code, database, programming language etc., as long as you get the job done and other services can depend on you.

This is how core legacy architecture looks liks



Core legacy situation: complexity

Microservices and Containers Sprawl – reproducing "core legacy situation"

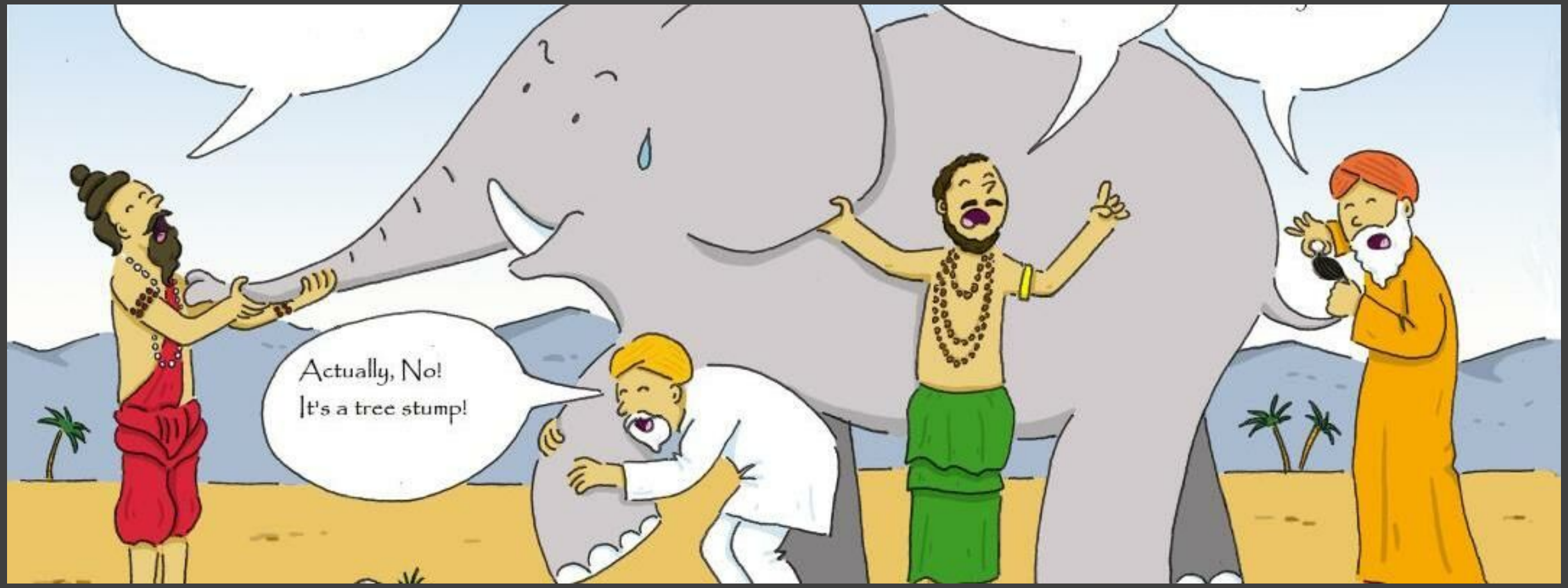
The solution: The new architect

Modern technology architecture

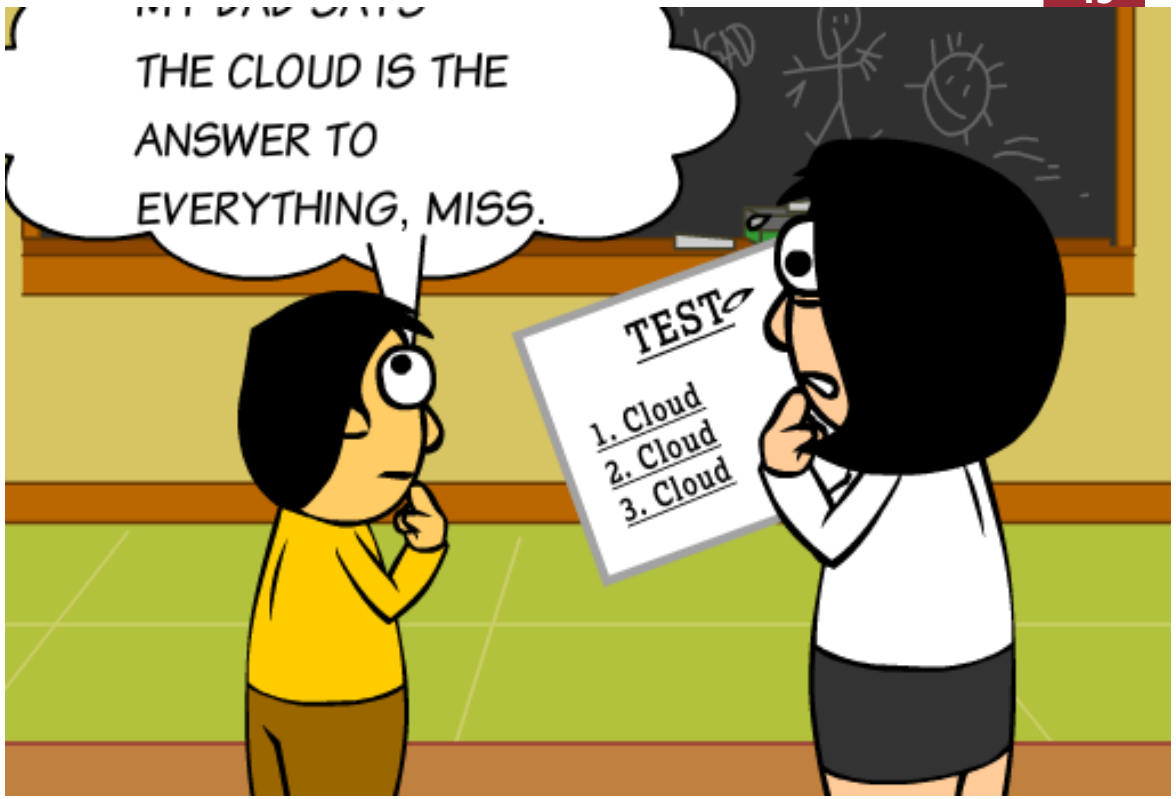
- Decentralized team
- Working with the teams
- Give services to all products\projects
- Keep track of the whole picture
- Updated with latest technology and its implications

Agenda

- COVID-19 IT market implications
- Infra. & Ops. trends, organization, SRE
- Dev. & architecture trends, Self Contained Systems (SCS), EDA (Event Driven), microservices sprawl
- Cloud vs. IT

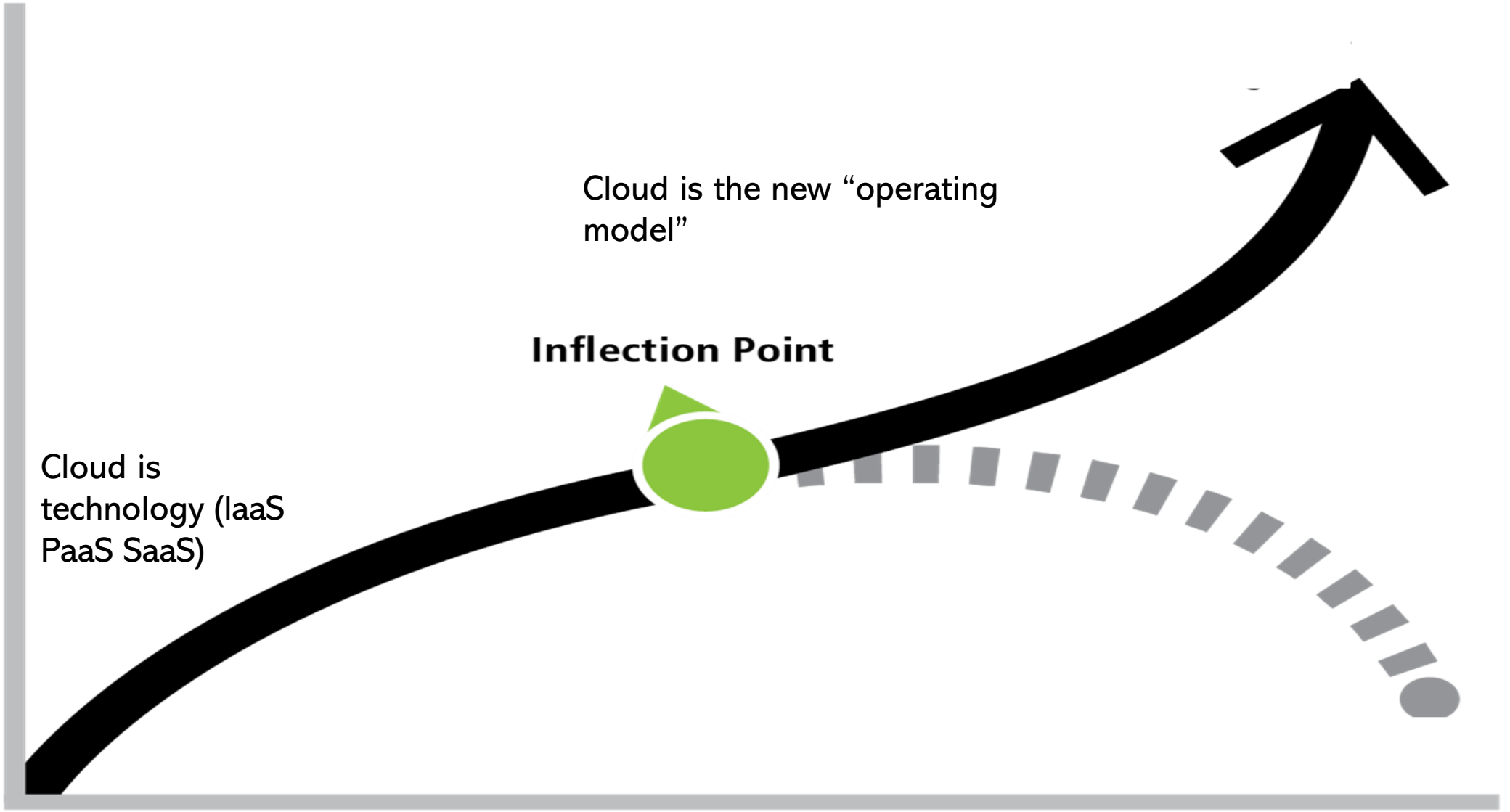


What is cloud?



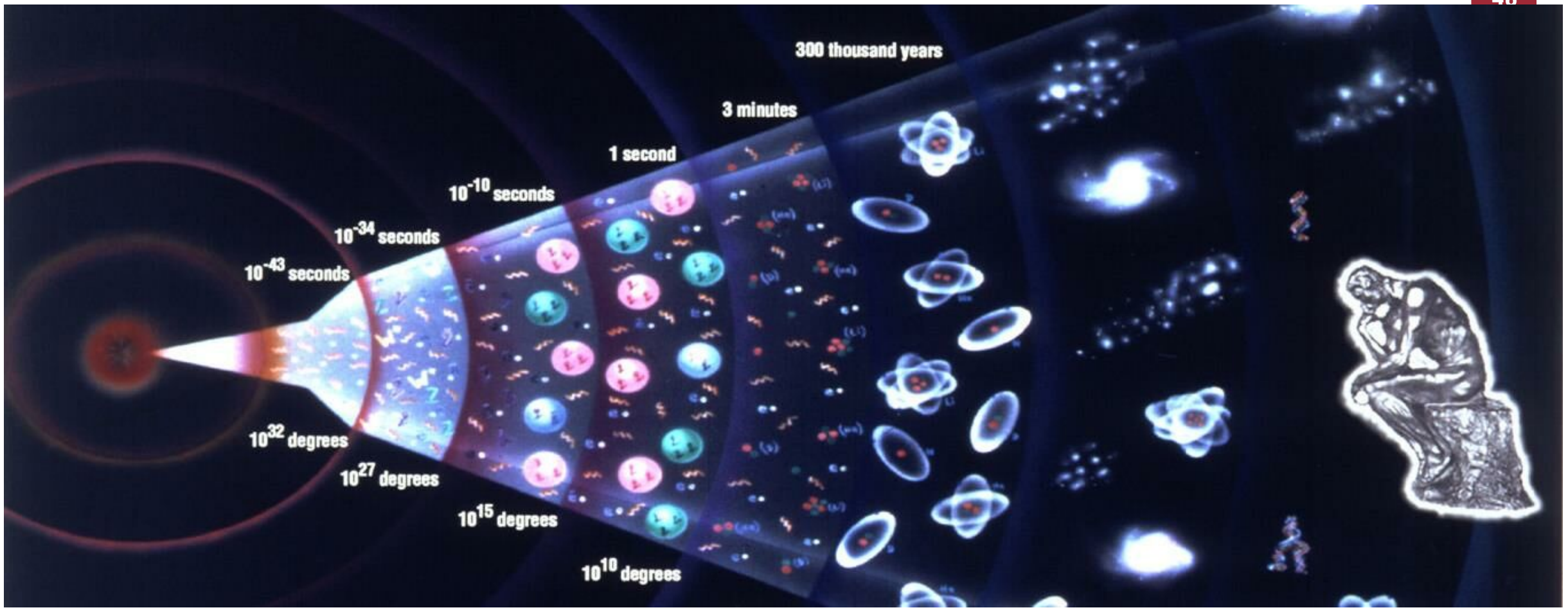
So what is cloud?

Cloud inflection point



Oracle cloud region in Israel with unique enterprise features:

- “Real” standard bare metal compute (100% CPU power – no “management agents”)
- Oracle Cloud's Hardware Root of Trust
- Maximum Security Zones lets customers denote enclaves within their Oracle Cloud Infrastructure (OCI) environments that have all security measures turned on by default.



Azure region in Israel

The END

Pini Cohen, CTO
pini@stki.info