Modern Architectures

From monolithic legacy applications to modern, reactive microservices

Technical University of Košice April 27th, 2017

SENACOR

Dr. Michael Menzel

Personal details

- Architect
- Diploma in Computer Science, University of Trier
- Doctorate at Hasso-Plattner-Institute, Potsdam
- Clients: Telecom, VW, Deutsche Postbank
- > 15 years of development experience

Area of expertise

- Software development and programming
- Large distributed systems architectures
- System integration, web service technologies and frameworks
- IT security & identity management

Example projects

- Technical Lead for intermediary business middleware services for the largest German retail bank
- Technical lead for a middleware authorization framework of a leading retail bank

SENACOR

Daniel Heinrich

Personal details

- Senior Developer
- M.Sc. Internet & Web Sciences, Hof University of Applied Sciences
- Clients: Airbus Defense & Space, Federal Employment Agency
- > 5 years of development experience

Area of expertise

- Software development and programming
- System integration, web service technologies and frameworks

Example projects

- Lead architect for a logistics support analysis tool of an European airplane project
- Architect for the placement and consultation system of the Federal Employment Agency

Senacor is a leading independent consultancy for Business-IT Transformation in Germany

Company

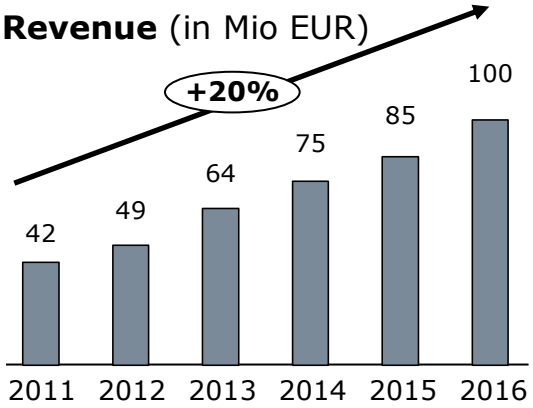
- Transformation planning incl. master planning, target IT-architecture and IT-roadmap
- Focused on strategic, long-term cooperation with relatively few clients within the scope of large transformation projects
- Industries: Financial Services, Insurance Business, Automotive, Telecommunication and Public sector

Key data

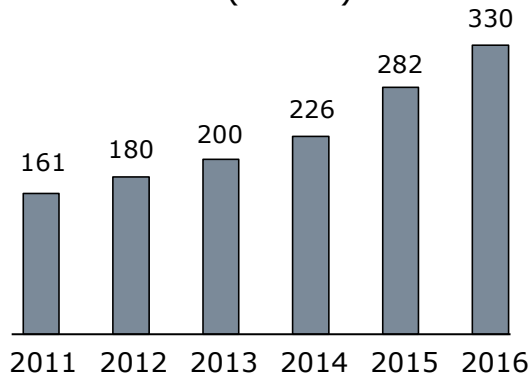
- **Founding:** 1999 in Nuremberg
- **Number of employees**: ~ 405, (330 consultants)
- **Revenue 2015:** 100 Mio.
- **Leadership**: Partnership
- **Shareholders:** Management, private investors

Revenue development

Revenue (in Mio EUR)



Consultants (in FTE)



Our Offices

- 9 offices in Germany, Austria and Slovakia
- International projects in German companies

SENACOR

Senacor offers end-to-end support for IT transformations triggered by a variety of business and technology needs

International expansion

Outsourcing transformation

Core system renewal

Post-Merger integration & consolidation

IT Transformation

Consulting

Business-IT Strategy

- Business IT Alignment
- Process and IT landscape analysis
- Portfolio management
- IT cost efficiency
- IT organization and processes
- IT governance
- Enterprise architecture

Planning

- Business and technical target architecture
- Master planning
- Business case analysis
- Migration strategy
- Sourcing
- Program and project scoping
- Portfolio management

Delivery

Management

- Program office/ Program management
- Architecture office
- Project review & rescue
- Provider management
- Risk and quality management

Implementation

- Functional specification
- IT specification
- Software design und testing
- Systems integration
- Offshoring management
- Project management/ technical lead

Optimization of business capabilities

IT efficiency transformation

B2B cross-company integration

SENACOR

Senacor supports large clients characterised by high IT complexity

Banking

Deutsche Bank
Hypo Real Estate GROUP
FMS Wertmanagement
ING DiBa
FIDUCIA
Xchanging
KFW
Postbank
Team!Bank
Raiffeisen Bank International
Haspa Hamburger Sparkasse
DAB bank
DZ BANK Zusammen geht mehr.
LBBW
COMMERZBANK
.comdirect
HypoVereinsbank UniCredit Group
Cortal Consors
Deka
Portigon Financial Services
Bank Audi

Logistik

Deutsche Post
DHL
Die Bahn
DB
GLS
Hapag-Lloyd

Automotive

DAIMLER
VW

Telco

vodafone
T

Öffentlicher Sektor

Bundesagentur für Arbeit

Versicherungen

AXA
VERSICHERUNGSKAMMER BAYERN
Allianz
Finanzgruppe
Gothaer
VHV VERSICHERUNGEN

Other

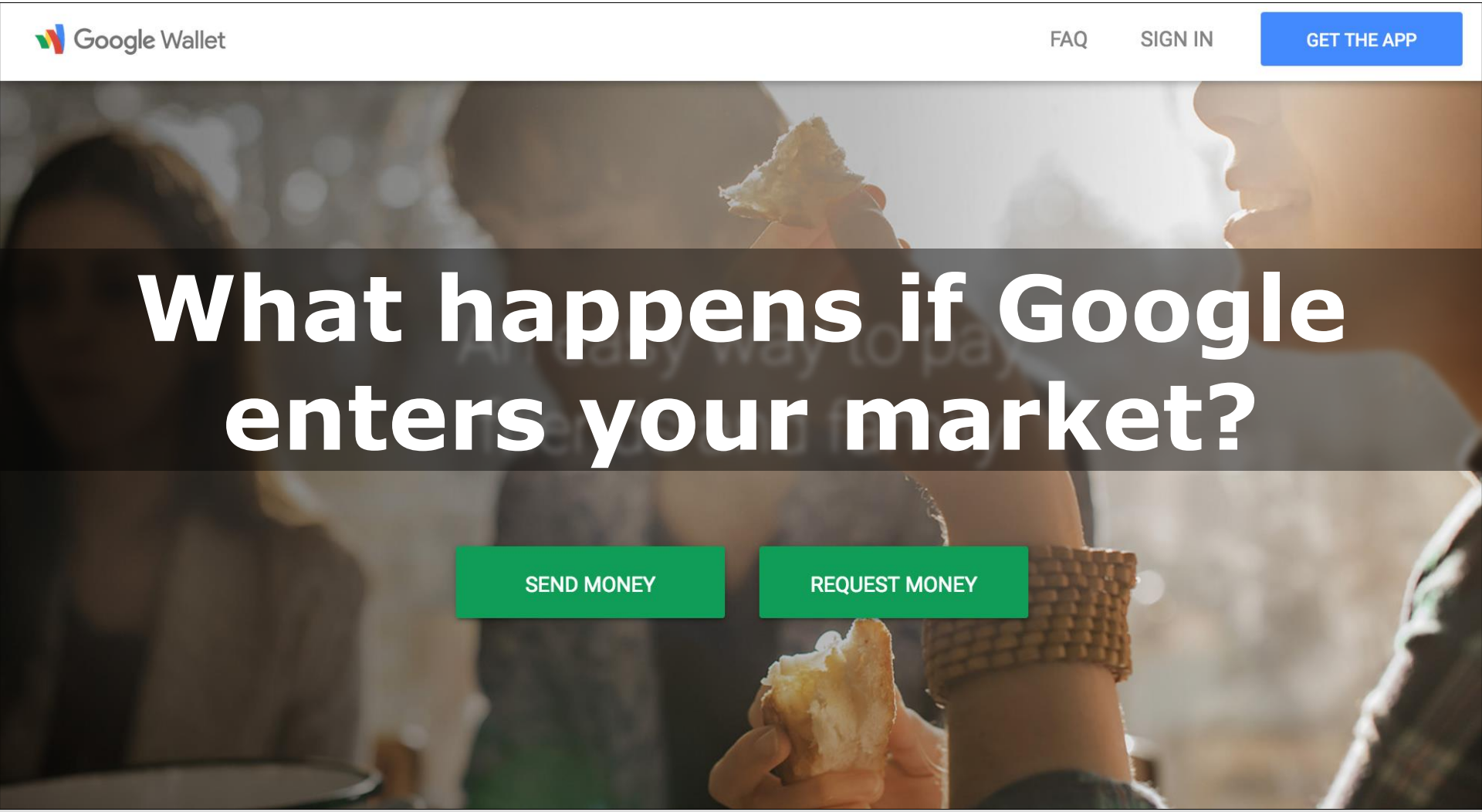
RWE
EnBW
ESPRIT
adidas GROUP
GfK
METRO Group

SENACOR

- **Software is eating the world**
- What is architecture?
- Status quo: Multi-Channel Architecture
- Digital-Ready Architecture
- Assessment and conclusion

SENACOR

Digitalization changes everything



What happens if Google enters your market?

SENACOR

Software is eating the world – and large corporations are struggling to adapt to the challenges

Who is the Disruptor?

SENACOR

Digitalization changes everything

Blue Chips

amazon payments

Google wallet

PayPal

Pay

MyWallet

T

Bank

FinTecs

PAGIDO

elopay

auxmoney

bitcoin.de

kash smart bezahlen

quirion

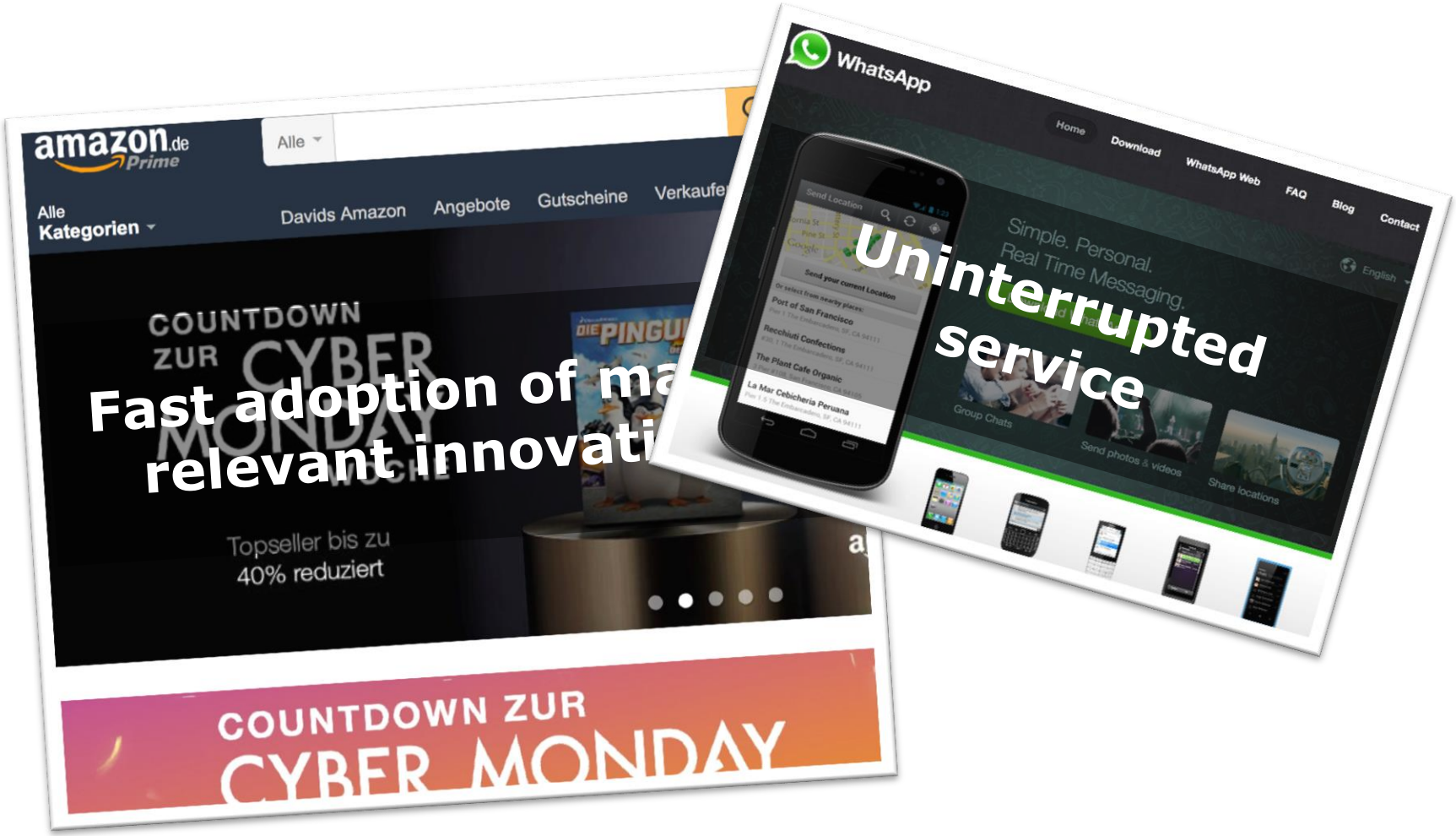
SENACOR

Digitalization changes everything

Fast adoption of market relevant innovations

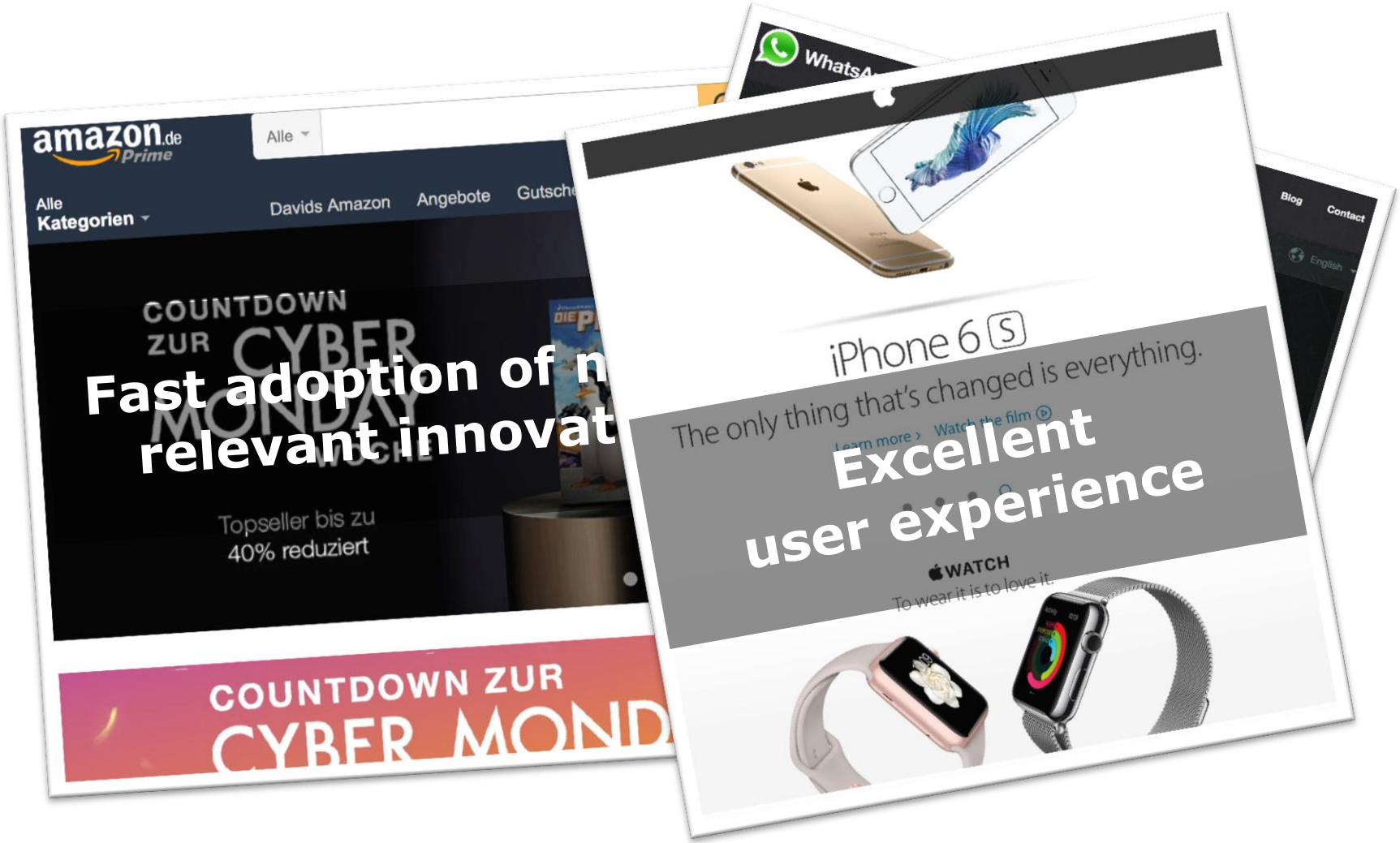
SENACOR

Digitalization changes everything



SENACOR

Digitalization changes everything



Fast adoption of n
relevant innovat

Excellent user experience

SENACOR

Digitalization changes everything

SENACOR

The "Digital Ready Architecture" requires a different approach and is based on a set of key assumptions/guidelines

Digital Ready Architecture

1. Embrace Change
   Fast delivery of ideas into production — Changeability
2. Elasticity and Scalability
   Cloud native applications — Elasticity
3. Embrace Failure
   Cope with failures; high availability — Resilience
4. Responsiveness and Reactivity
   Excellent user experience from end-to-end — Reactivity

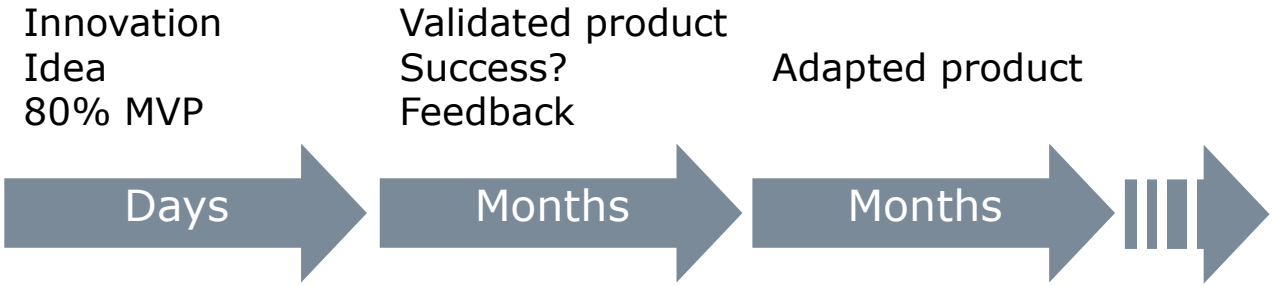
SENACOR

Innovation is key for surviving the market changes

Changeability

Traditional corporations lack a fast feedback loop and innovation chain

Disruptive market forces release ideas faster and more frequently



SENACOR

We want to stay up, even with high load

Elasticity

How much load will the Christmas Season put on my systems?

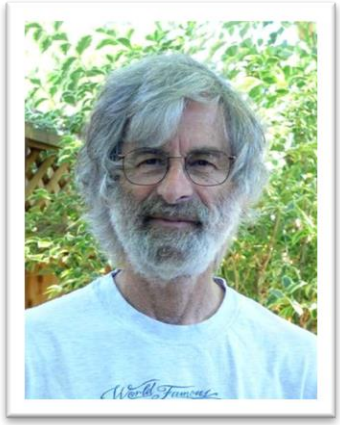
Dynamic Scalability is the key

- ✓ transparent scale-out
- ✓ dynamic provisioning of resources
- ✓ Cloud-readiness

SENACOR

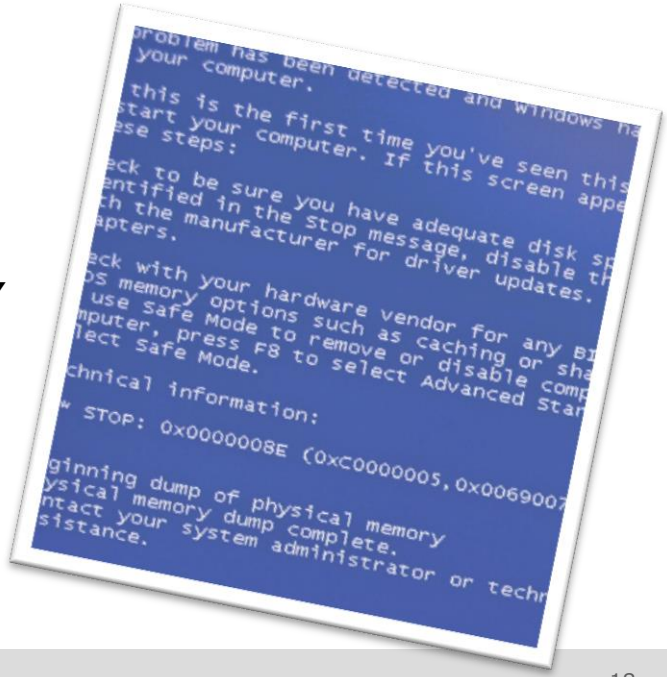
The system should stays responsive in the face of failure with a graceful degradation of service if required

Resilience



A distributed System is one in which the failure of a computer you did not even now existed can render your own computer unusable.

- **Leslie Lamport**



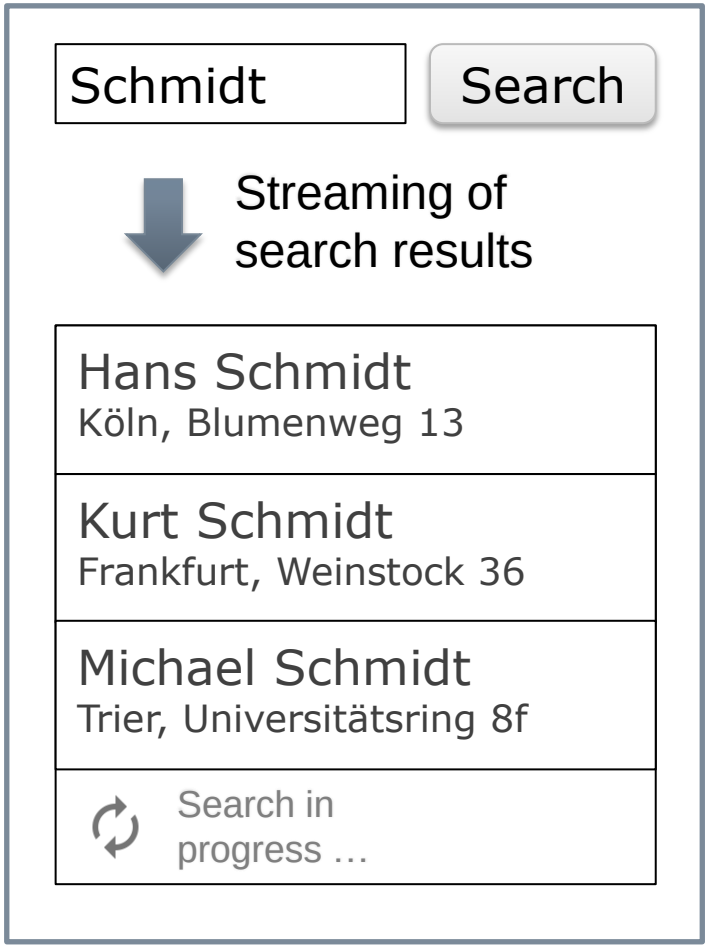
Crash Failure, Omission Failure, Timing Failure, Response Failure, Byzantine Failure

➔ Do not try to avoid Failures. Embrace them.

SENACOR

Applications should be highly interactive enabling an excellent user experience from end-to-end

Reactivity



"Nobody likes to wait and look at an AJAX loading indicator"

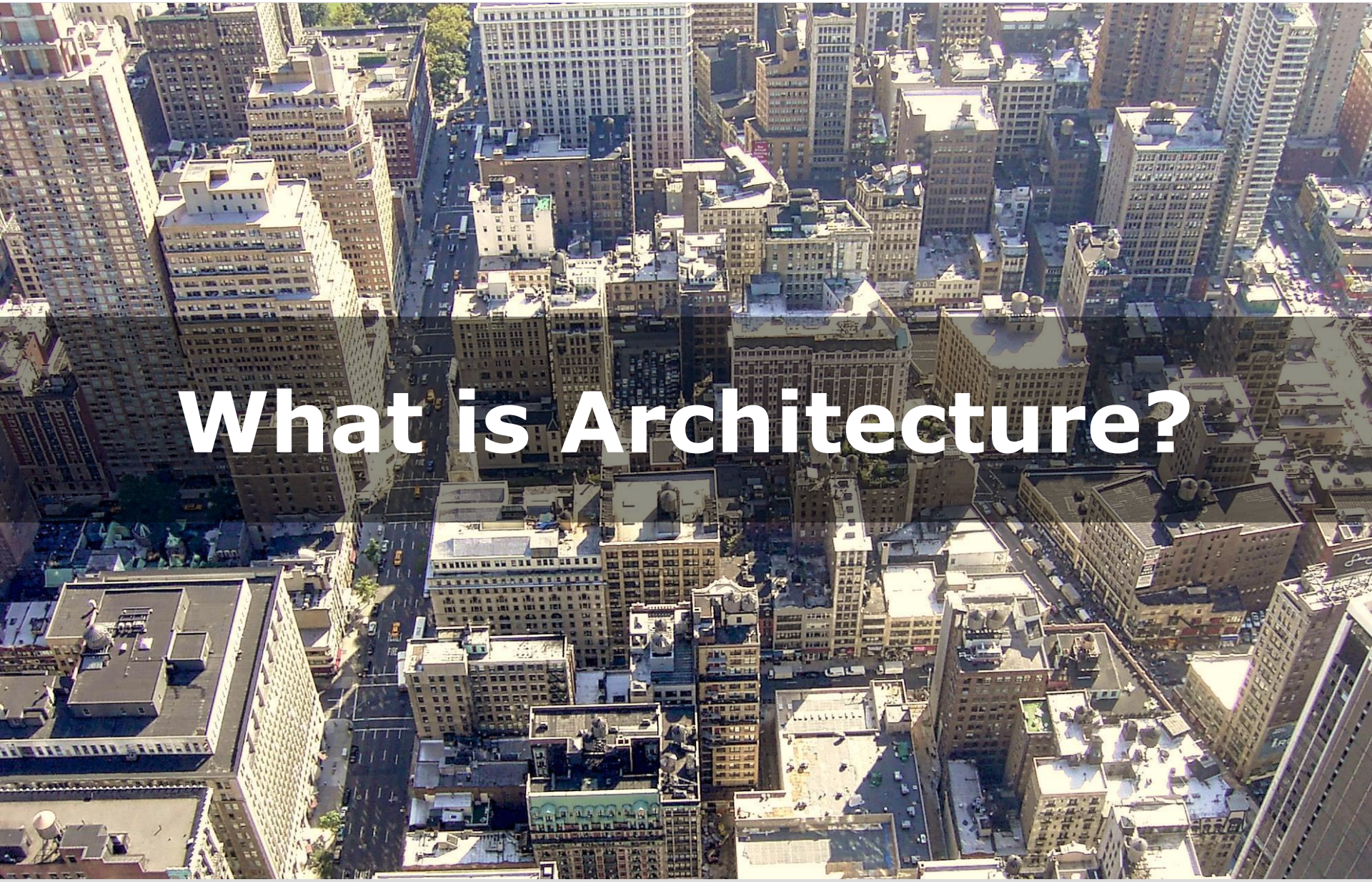
Reactive systems are push-based:

- Asynchronous data processing
- Streaming of information

SENACOR

- Software is eating the world
- **What is architecture?**
- Status quo: Multi-Channel Architecture
- Digital-Ready Architecture
- Assessment and conclusion

SENACOR



What is Architecture?

SENACOR

Three different abstraction levels for architecture guide the development of software applications

- Architecture at the level of whole domain areas
- Concepts include business areas and organizations such as: Loan, HR, trading, CRM
- Technical details in general not a primary concern

Focus of this presentation:

- Architecture at the level of systems and system of systems
- Concepts include protocols, replication, intra-domain messaging
- Typically one would find patterns such as SOA, ESB, CQRS, event sourcing,…

- Architecture at the level of deployable units
- Concepts include the actual design of services
- Typical abstractions and technologies include: Spring, EJB, BCE, reactive design patterns, SPA and concreate programming languages

SENACOR

But in general terms: architecture is basically about the way people collaborate

Yes, Teamwork

SENACOR

Conway's Law

…organizations which design systems … are constrained to produce designs which are ***copies of the communication structures*** *of these organizations*

- M. Conway

SENACOR

Teams will develop focusing on their area of expertise

Web Frontend Team	Frontend
Services Team	Services
Database Team	Databases
Ops Team	Application Servers

SENACOR

Loan gets built

Project Loan

Web Frontend Team

Loan UI

Services Team

Loan Service

Database Team

Loan Database

Ops Team

Loan App. Serv.

SENACOR

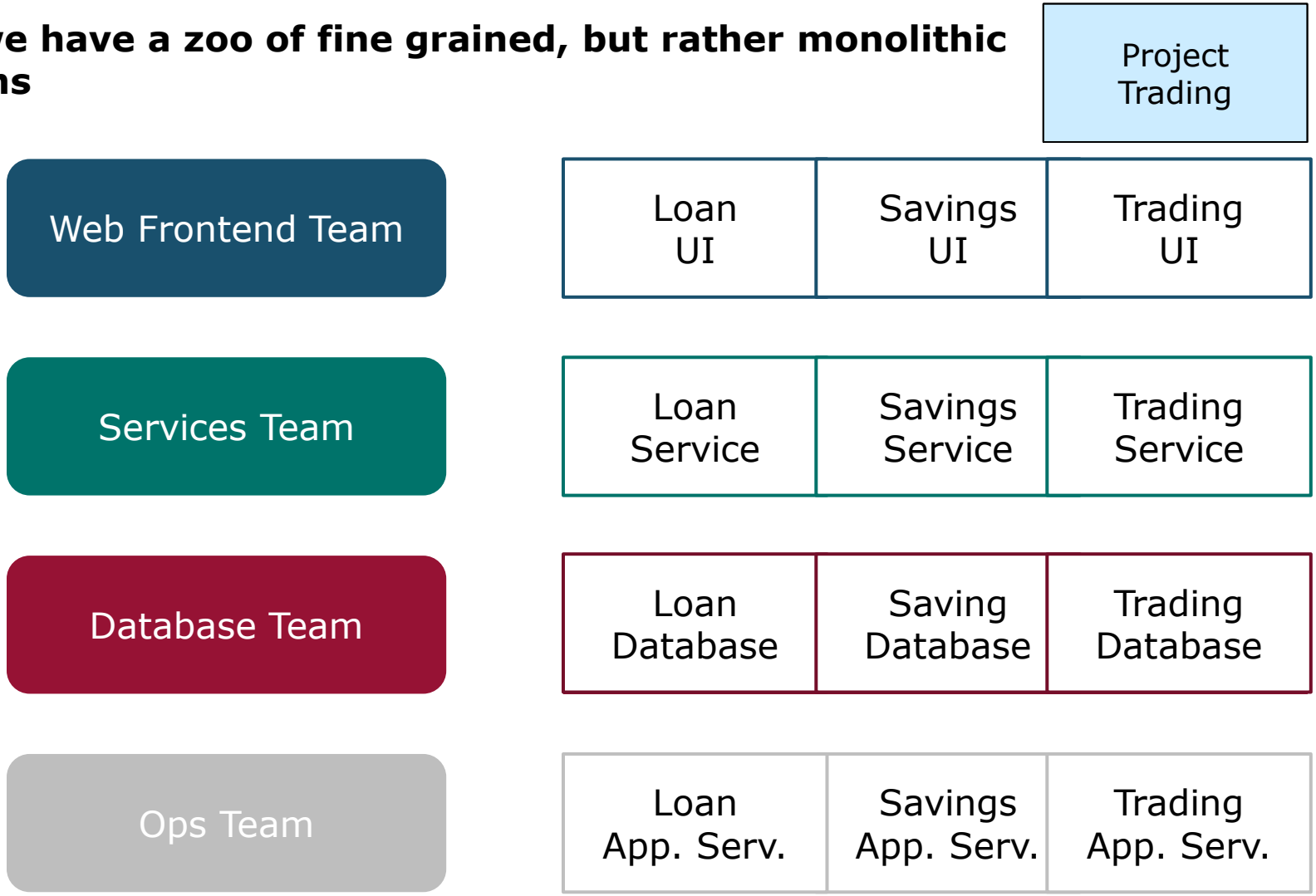
Let’s build some savings systems

Project Savings

Web Frontend Team	Loan UI	Savings UI
Services Team	Loan Service	Savings Service
Database Team	Loan Database	Saving Database
Ops Team	Loan App. Serv.	Savings App. Serv.

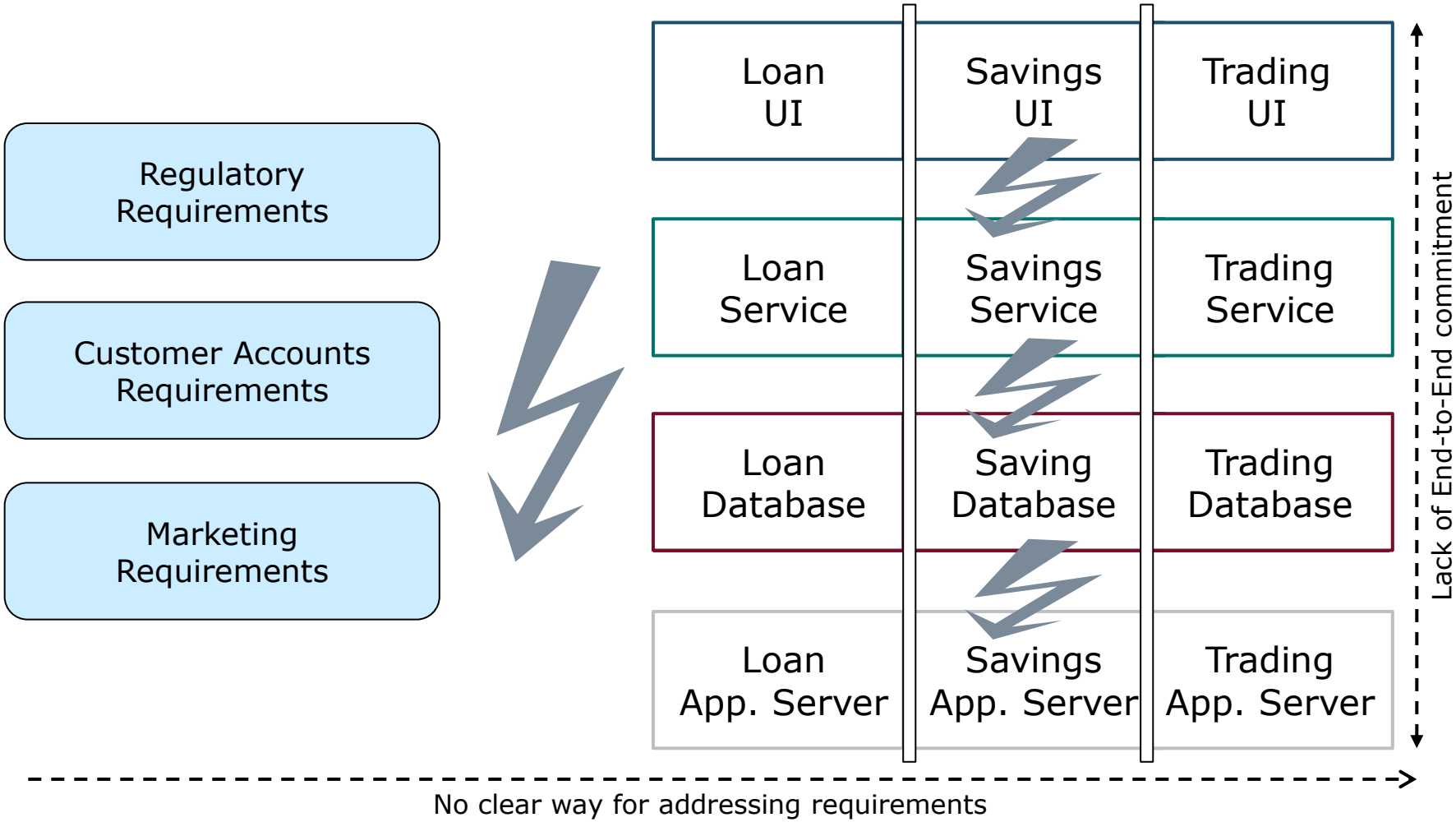
SENACOR

Now we have a zoo of fine grained, but rather monolithic systems



SENACOR

Businesses do no not work this way! Cross-cutting concerns and other requirements need a complex change process and are risky to develop

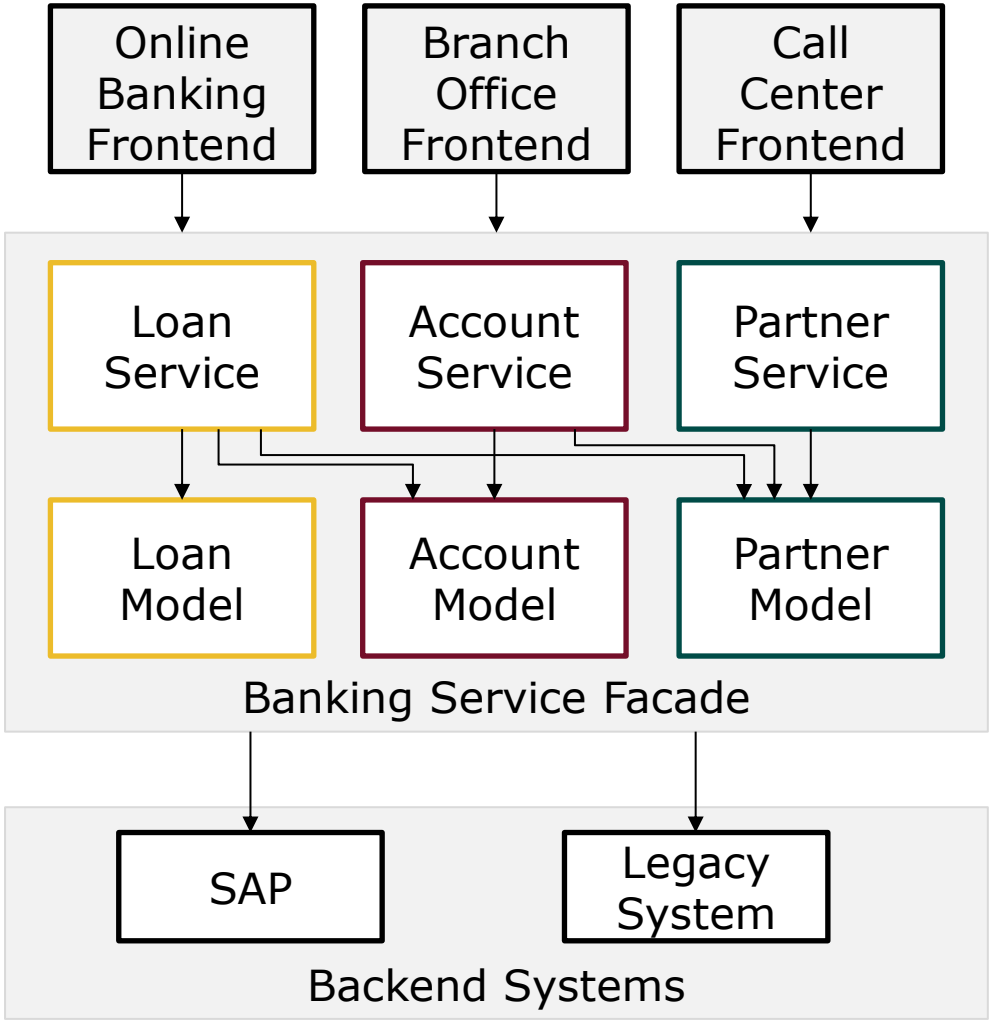


SENACOR

- Software is eating the world
- What is architecture?
- **Status quo: Multi-Channel Architecture**
- Digital-Ready Architecture
- Assessment and conclusion

SENACOR

Status-Quo: Multi-Channel Architecture to foster service reuse

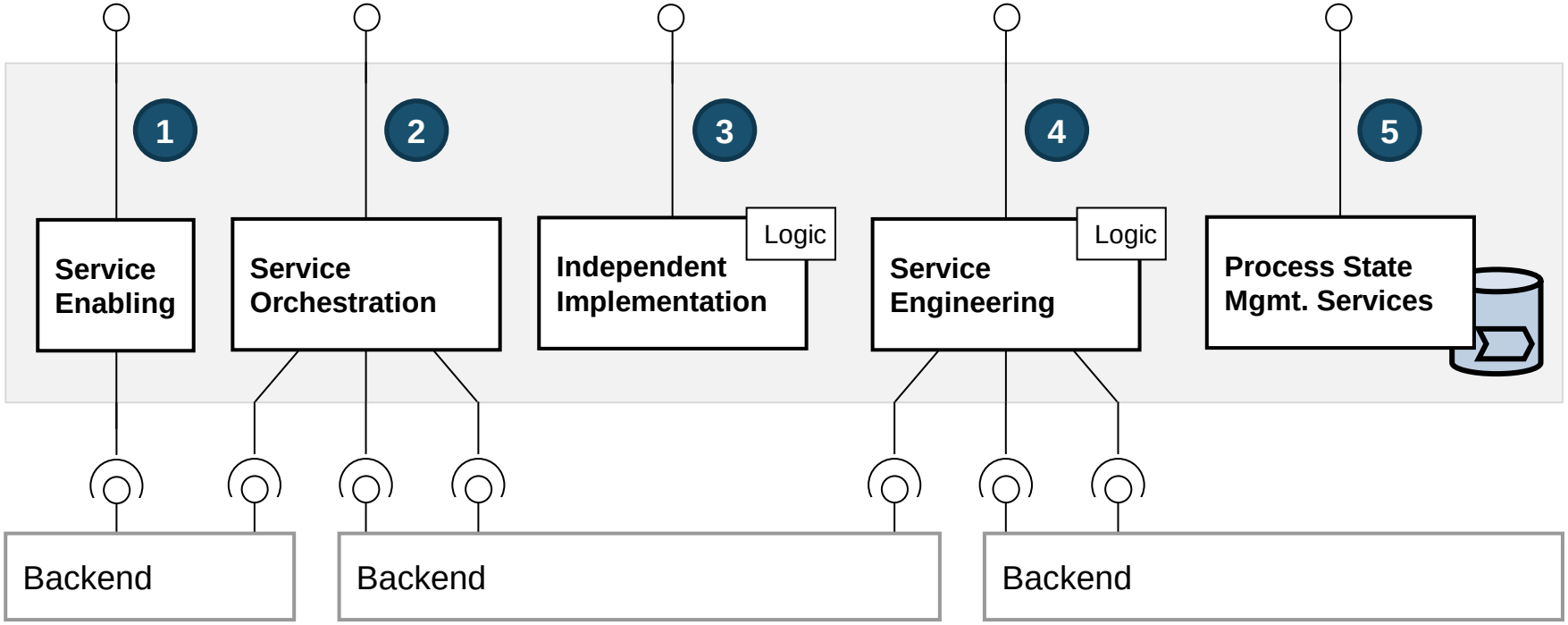


Service Facade

- Provision of channel-independent services
- Harmonized Interfaces Technology
- Maximize reuse
- Canonical Data Model / Shared Entities

SENACOR

Facade Services integrate backend services or provide custom implementations



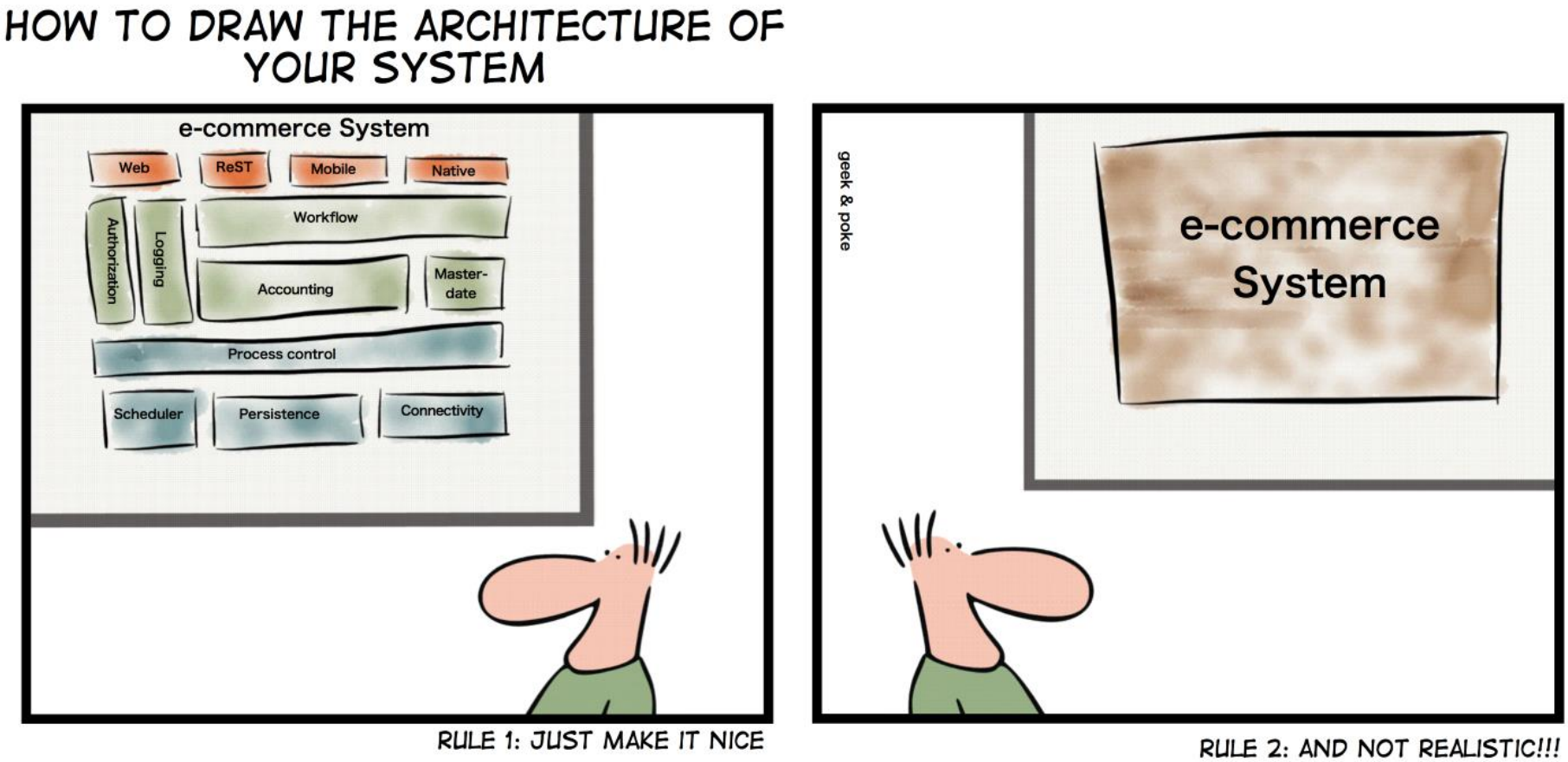
SENACOR

6 Monoliths.
600 Services. 25 Clients. 45 Backends.

Grown code base since 1995.

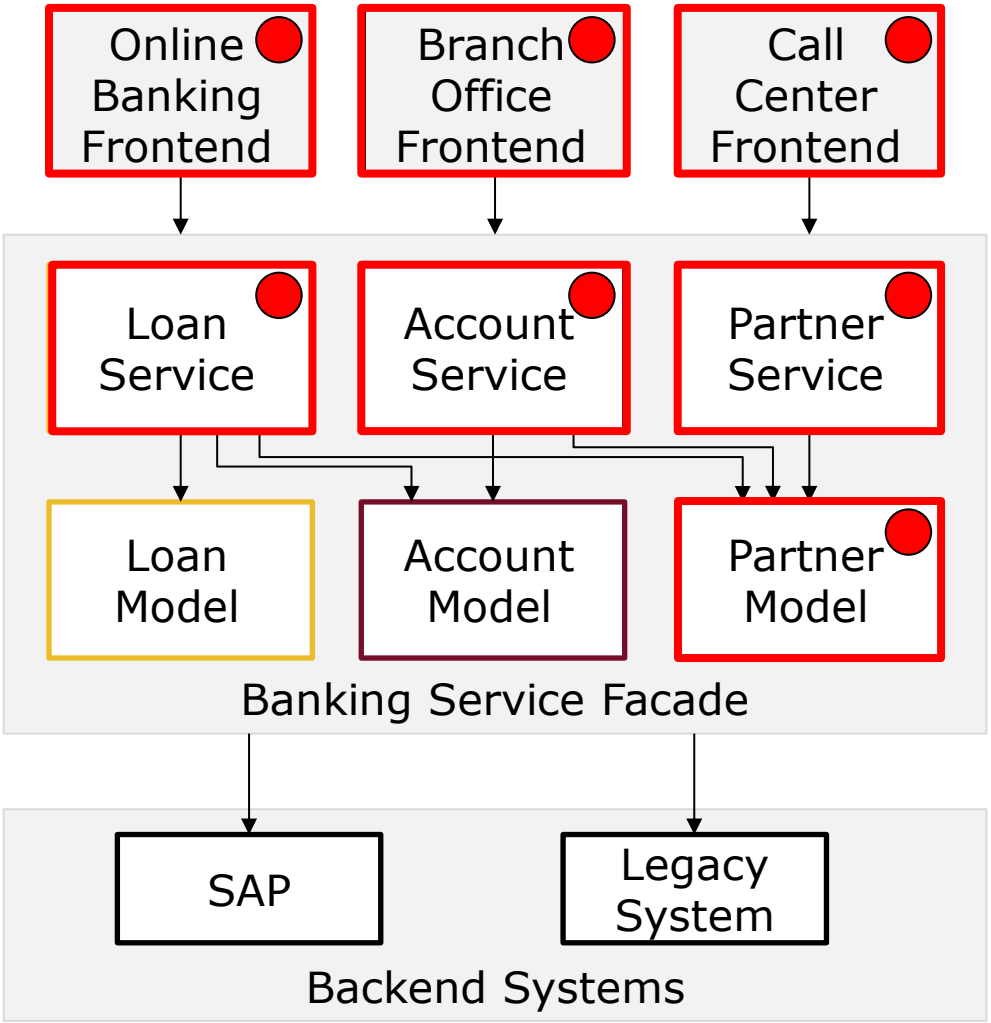
(Someone remembers J2EE 1.2?)

SENACOR


HOW TO DRAW THE ARCHITECTURE OF YOUR SYSTEM
e-commerce System
Web
ReST
Mobile
Native
Authorization
Logging
Workflow
Accounting
Master-date
Process control
Scheduler
Persistence
Connectivity
RULE 1: JUST MAKE IT NICE
geek & poke
e-commerce System
RULE 2: AND NOT REALISTIC!!!

SENACOR

Although the applications are highly componentized, at runtime these form a monolithic structure that is rather difficult to change in key areas



Enhancing the Business Partner

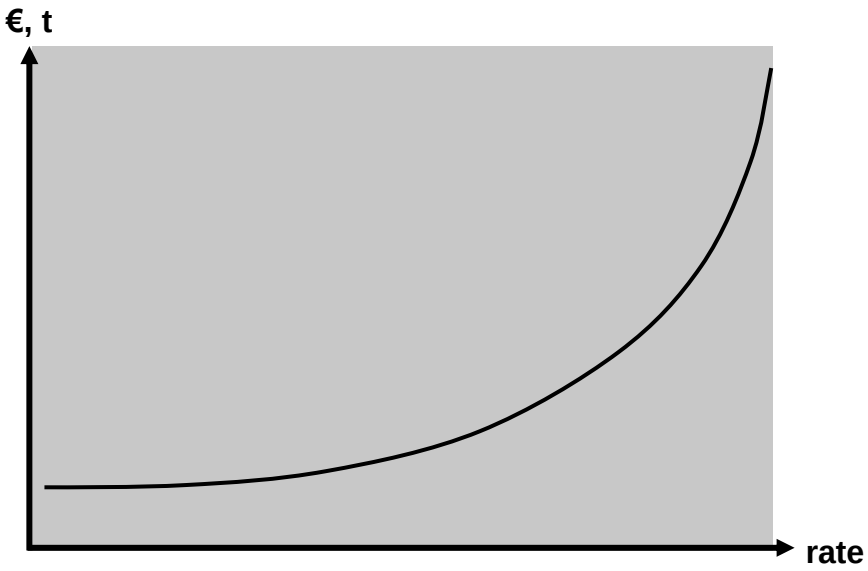
Which Components are affected?

SENACOR

Monolithic Applications lead to an increased effort for modifications

Effort for modifications

- Time
- Cost
- Risk
- Technical knowledge



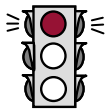
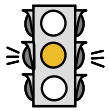
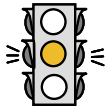
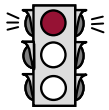
Application complexity

- Lines of Code
- Number of Interfaces/Classes/Services
- Libraries
- Dependencies to backend systems
- Startup logic

SENACOR

The multi-channel-architecture does not meet the requirements of an Digital Ready Architecture

	Evaluation
Embrace Change	▪ Effects of changes tend to ripple through the system ▪ Incompatible changes are risky and error prone
Elasticity and Scalability	▪ Scalability is usually achieved by clustering and scaling vertically in advance ▪ Fine grained scalability is in generally hard to achieve
Embrace Failure	▪ The reuse of systems lead to a brittle deployment monolith ▪ Measures to avoid reduced availability lead to complex and costly solutions
Responsiveness and Reactivity	▪ Typical patterns are blocking system calls and distributed transactions ▪ Achieving true reactivity and responsiveness is unlikely

SENACOR

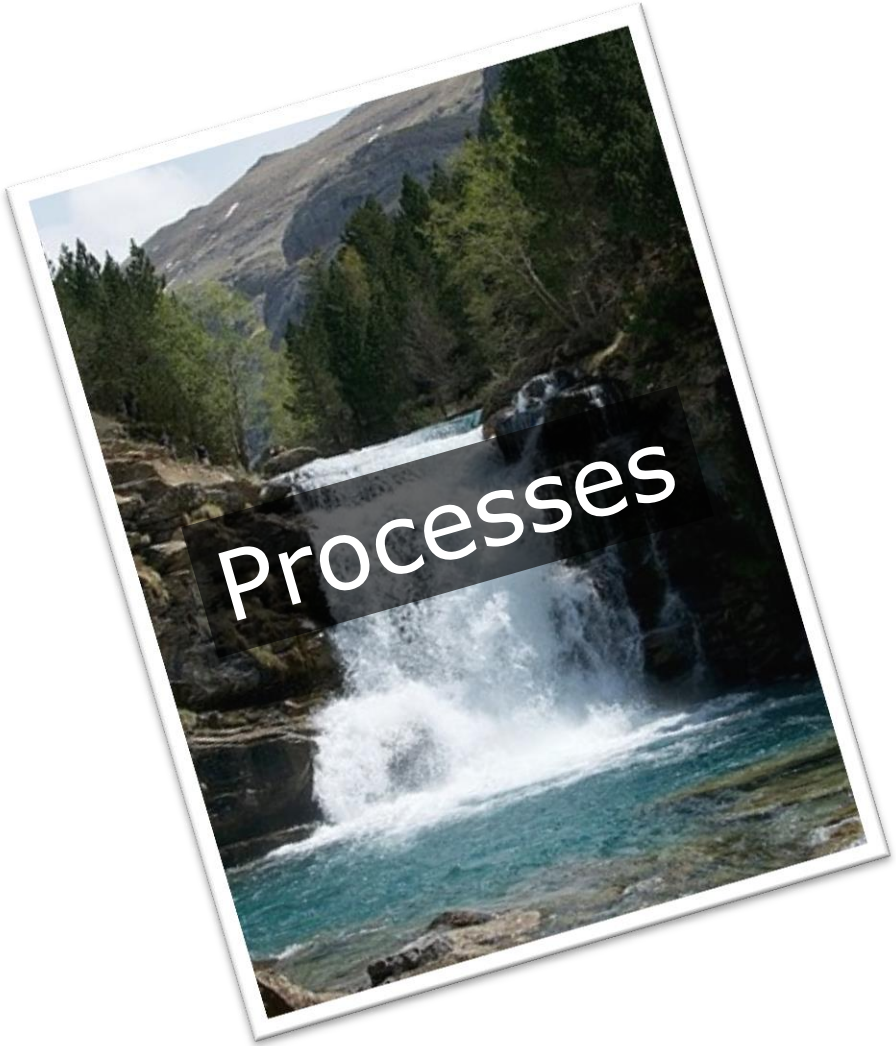
Multi-Channel Architecture does not encourage innovation

What‘s next?

SENACOR

- Software is eating the world
- What is architecture?
- Status quo: Multi-Channel Architecture
- **Digital-Ready Architecture**
- Assessment and conclusion

There are four primary areas that need to be changed, in order to cope with the challenges of today's realities



There are four primary areas that need to be changed, in order to cope with the challenges of today's realities

There are four primary areas that need to be changed, in order to cope with the challenges of today's realities

There are four primary areas that need to be changed, in order to cope with the challenges of today's realities

SENACOR

The "Digital Ready Architecture" requires a different approach and is based on a set of key assumptions/guidelines

Digital Ready Architecture

1. Embrace Change
   Fast delivery of ideas into production — Changeability
2. Elasticity and Scalability
   Cloud native applications — Elasticity
3. Embrace Failure
   Cope with failures; high availability — Resilience
4. Responsiveness and Reactivity
   Excellent user experience from end-to-end — Reactivity

SENACOR

The "Digital Ready Architecture" requires a different approach and is based on a set of key assumptions/guidelines

Digital Ready Architecture

1. Embrace Change
   Fast delivery of ideas into production — Changeability
2. Elasticity and Scalability
   Cloud native applications — Elasticity
3. Embrace Failure
   Cope with failures; high availability — Resilience
4. Responsiveness and Reactivity
   Excellent user experience from end-to-end — Reactivity

SENACOR

Why to embrace change

We want to be able to change often and quickly.

The reason for agile practices is to embrace change.

Agile is all about small development iterations.

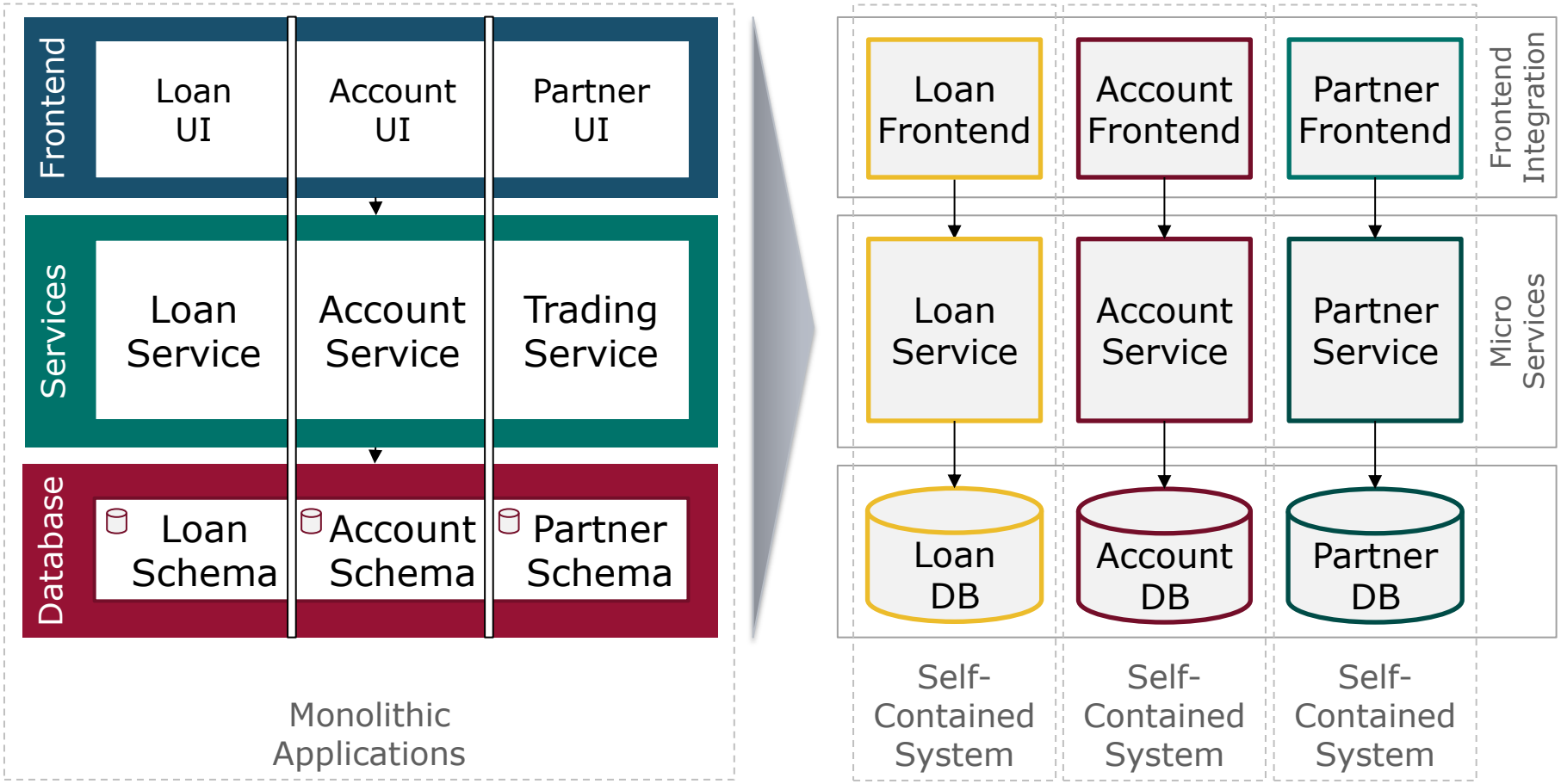
agilemanifesto.org

Benefits of small iterations

- Deliver features faster to the customer
- More controll over the direction of the project
- Don‘t waste money by investing in features nobody wants

SENACOR

How to embrace change: Vertical decomposition of monolithic applications in independent services in order to decouple releases



Release half a year

Release monthly **Release quarterly** **Release 2-weekly**

SENACOR

Vertical teams are required to deliver fast, dynamic and highly decomposed applications

Key principles

Vertical Feature Team

- Analysis, development, test, operation
- Reduced communication overhead / 2 Pizza Team

Organization leads to architecture

- Change your organization and your architecture will follow suit

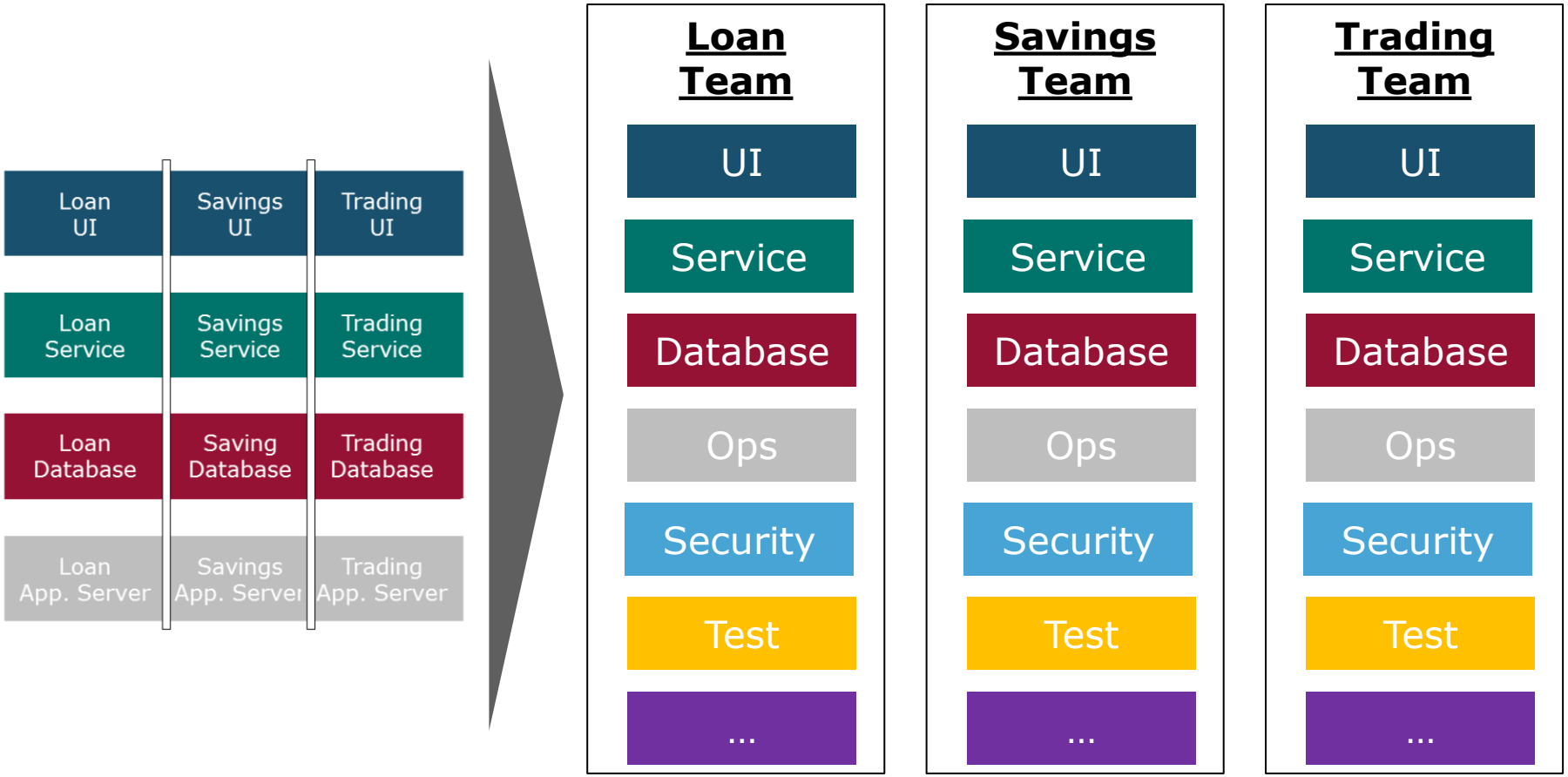
Products not Projects

- Development effort is focused on bounded contexts
- Clear separation of concerns

- Identify bounded-contexts based on business capabilities
- Long-lived responsibility leads to commitment and higher quality
- Systems are easier to understand, maintain, extend
- Communication is not organized around processes

SENACOR

Restructuring teams into vertical feature teams leads to end-to-end commitment



SENACOR

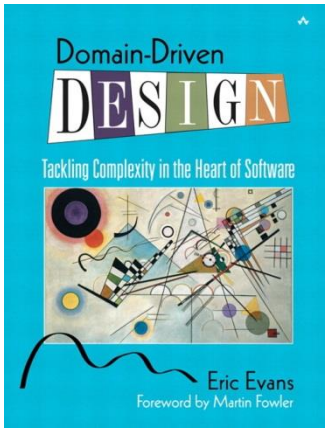
Inverse Conway Maneuver

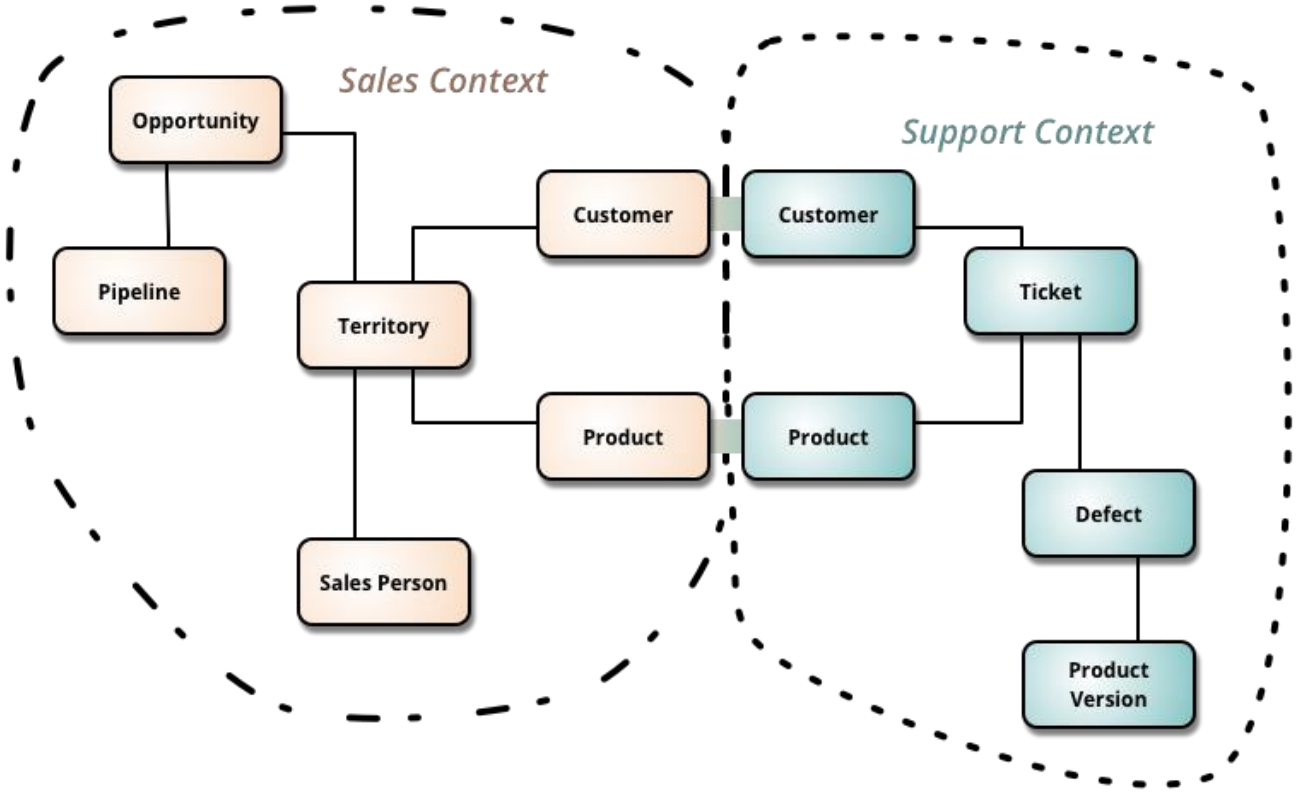
Evolve your team and organizational structure to promote your desired architecture. Ideally your technology architecture will display isomorphism with your business architecture.

https://www.thoughtworks.com/radar/techniques/inverse-conway-maneuver

SENACOR

Bounded-context: autonomous components, with own unified domain model and own ubiquitous language

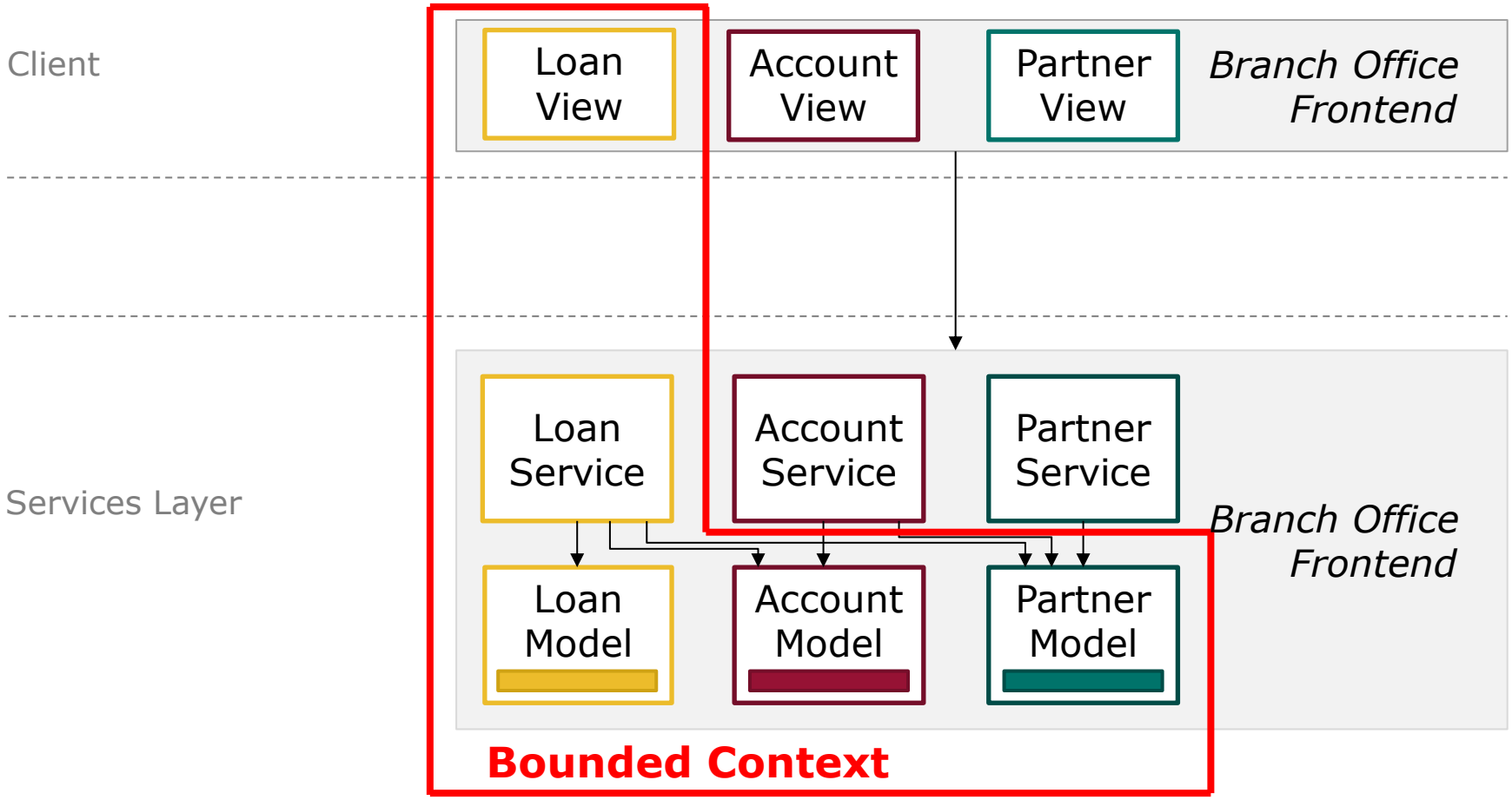




http://martinfowler.com/bliki/BoundedContext.html

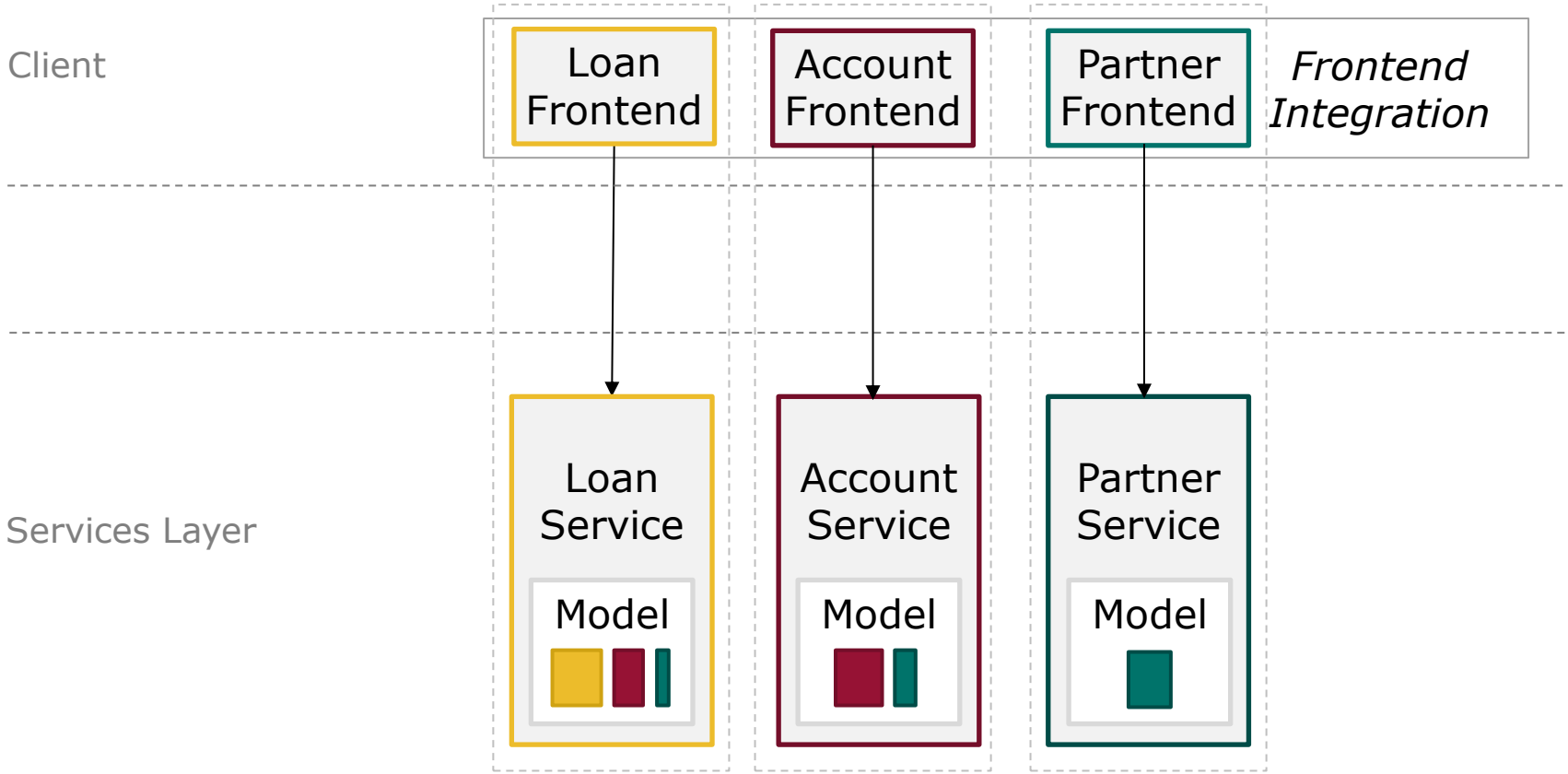
SENACOR

Multi-Channel-Architecture: Bounded Context of Loan Requirements spread over model classes



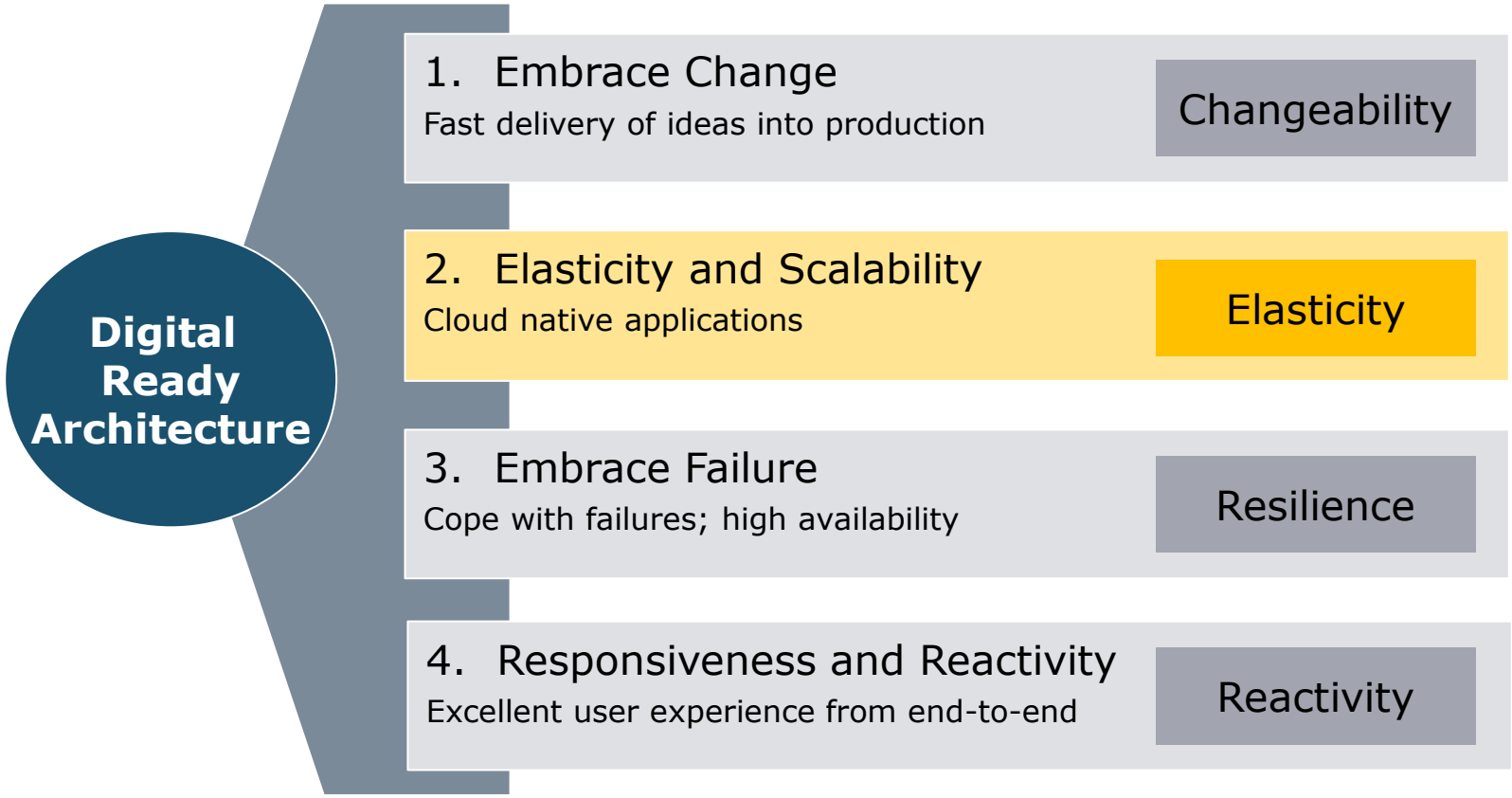
SENACOR

Vertical decomposition of services is based on copy & own; although certain key functions may be shared if needed



SENACOR

The "Digital Ready Architecture" requires a different approach and is based on a set of key assumptions/guidelines



SENACOR

Why being elastic

Scalability

- Being able to **scale up/out**
- Handle future needs

Elasticity

- Reacting to **changing demand** by scaling **bigger or smaller**
- Handle **current** needs

Being able to handle peaks

Avoid cost for unneeded infrastructure

SENACOR

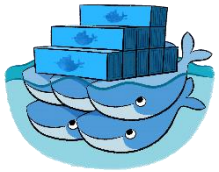
How to be elastic

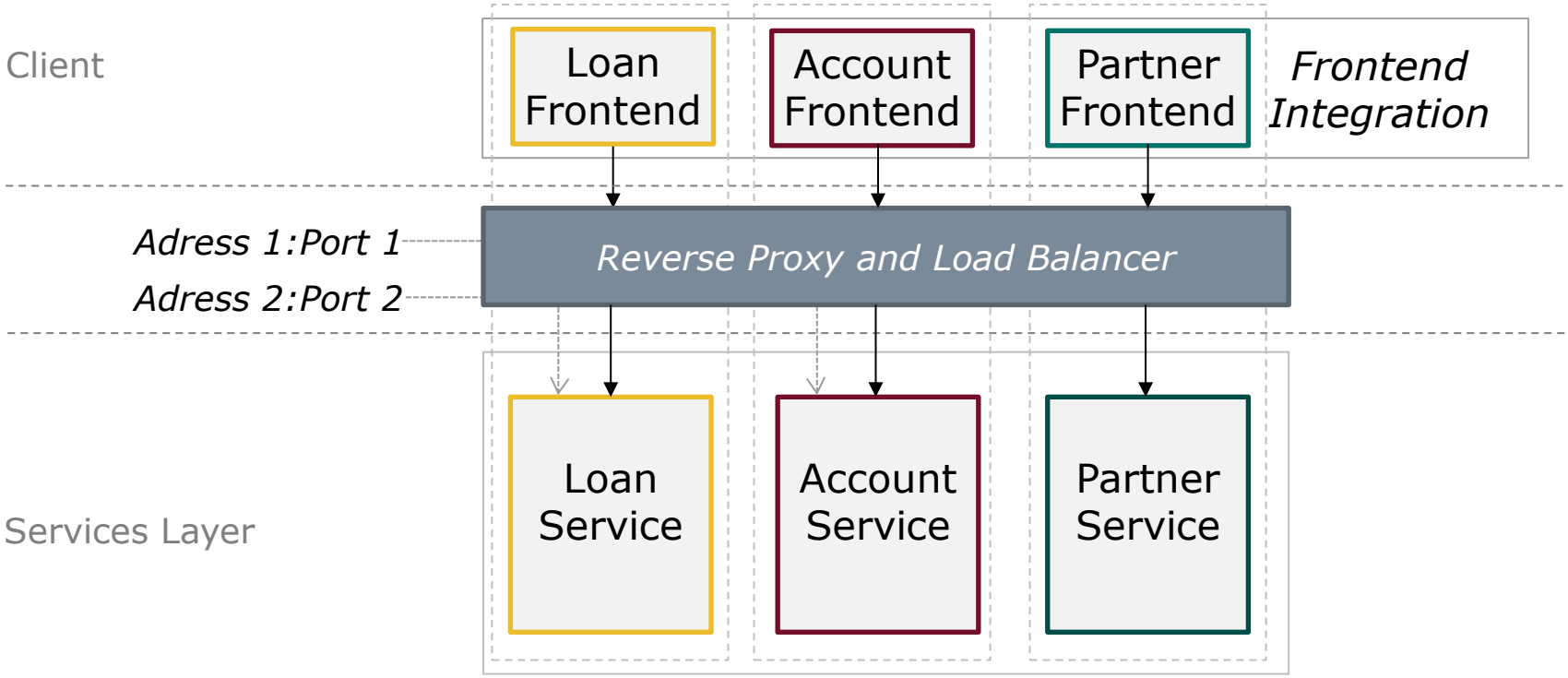
Immutable

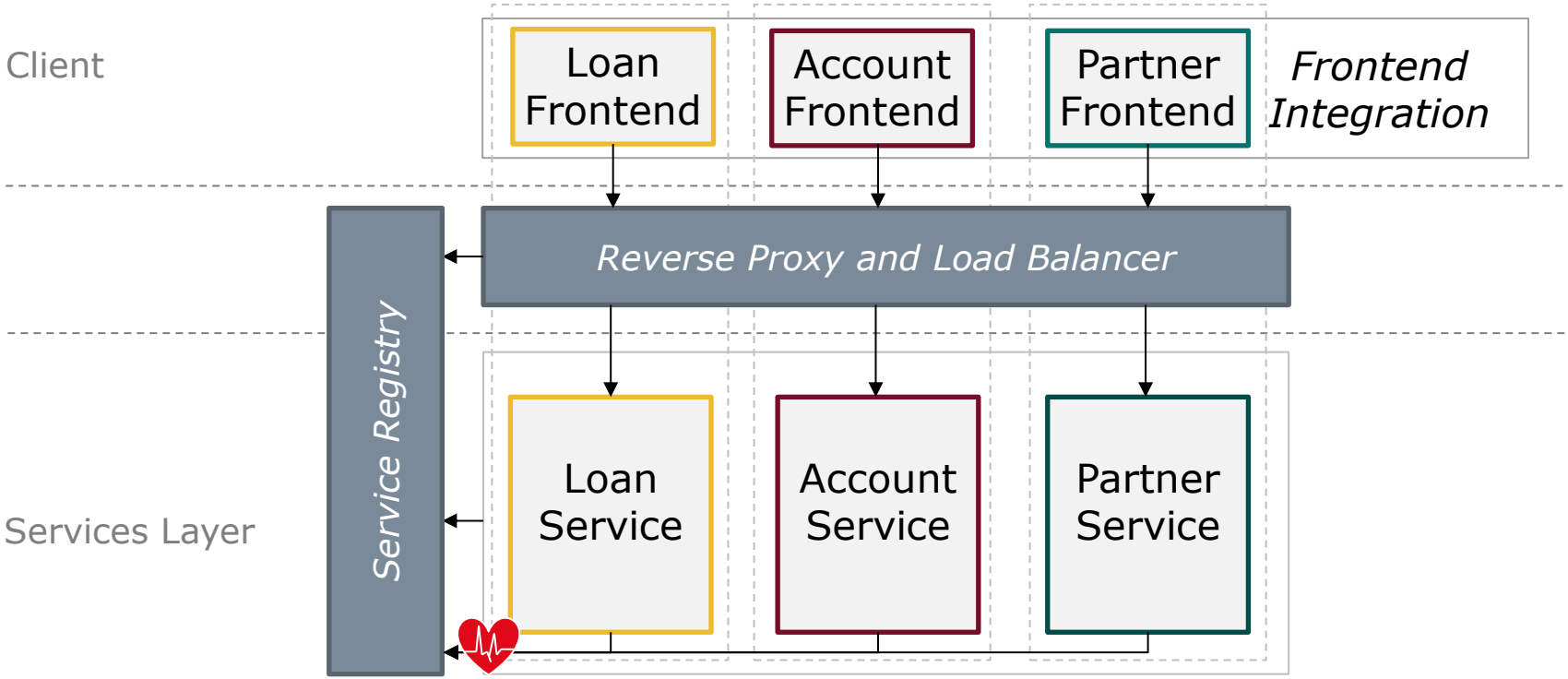
- immutable infrastructure, i.e. Docker container
- stateless services (with external session store)

Automatic

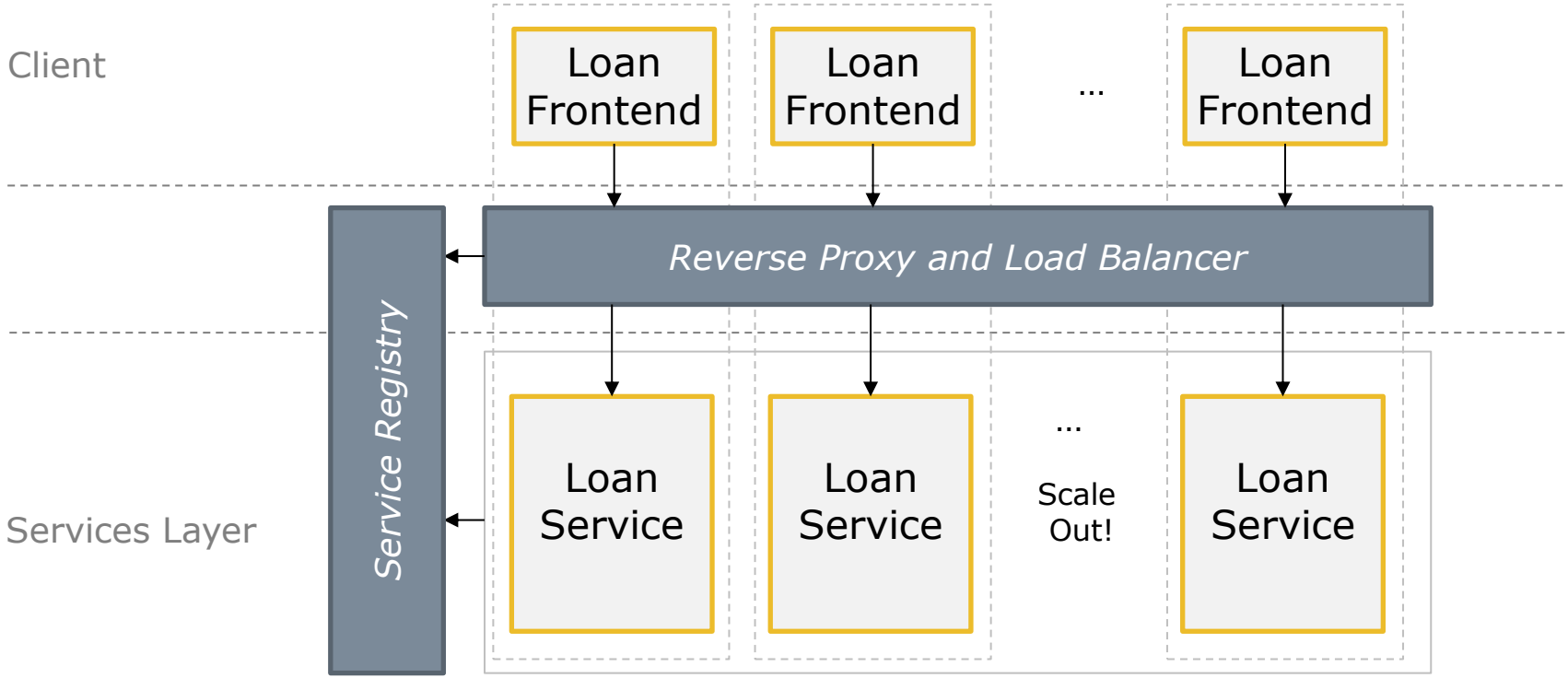
- auto-scaling, i.e. Kubernetes, Mesos Marathon
- reverse-proxy and load balancer
- service registry
- health checks




Client
Loan Frontend
Account Frontend
Partner Frontend
Frontend Integration
Adress 1:Port 1
Adress 2:Port 2
Reverse Proxy and Load Balancer
Services Layer
Loan Service
Account Service
Partner Service

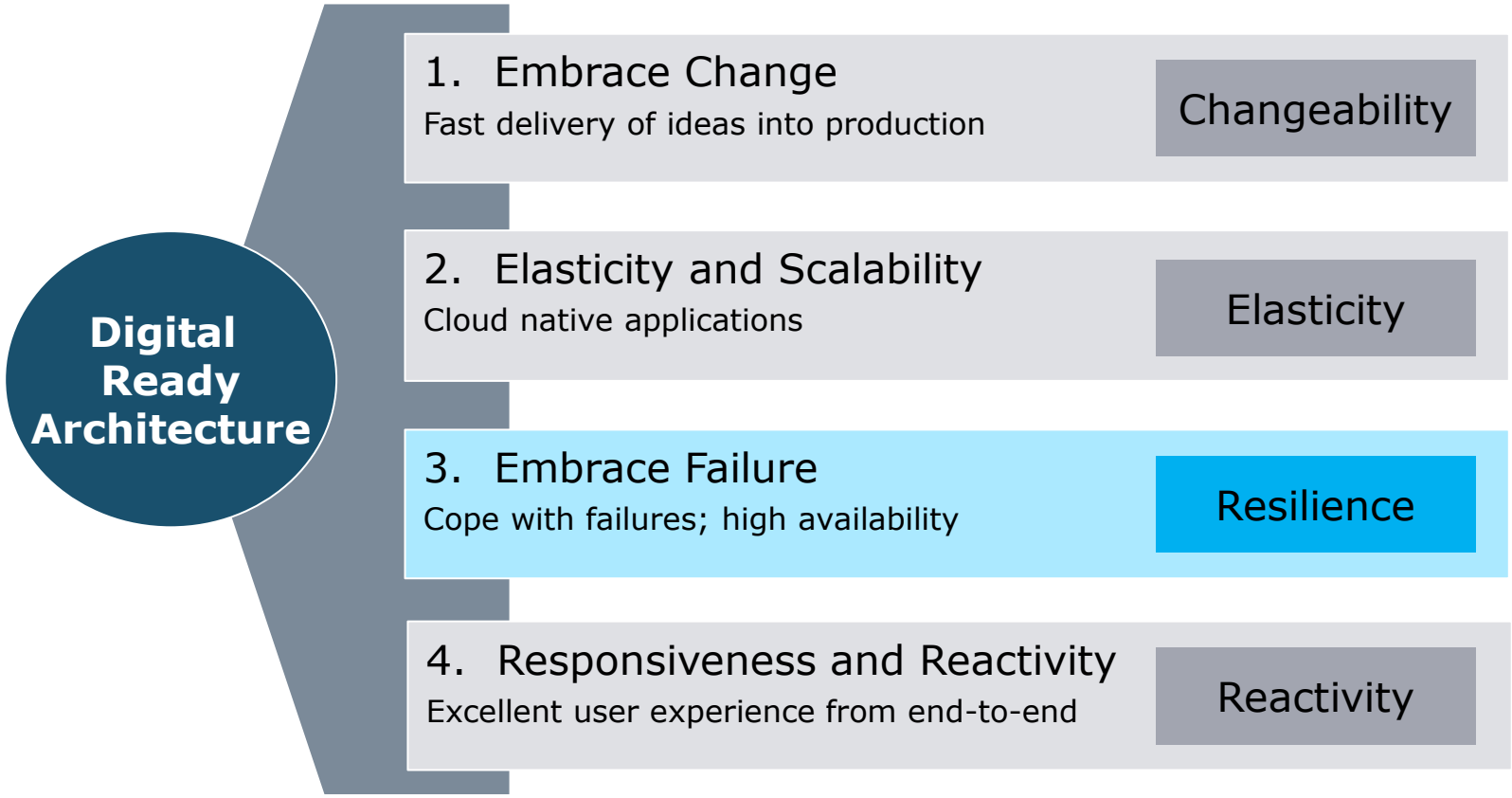


- Service registration on start up
- Heart beat checks for crashed services


Client
Loan Frontend
Loan Frontend
...
Loan Frontend
Reverse Proxy and Load Balancer
Service Registry
Services Layer
Loan Service
Loan Service
...
Scale Out!
Loan Service

SENACOR

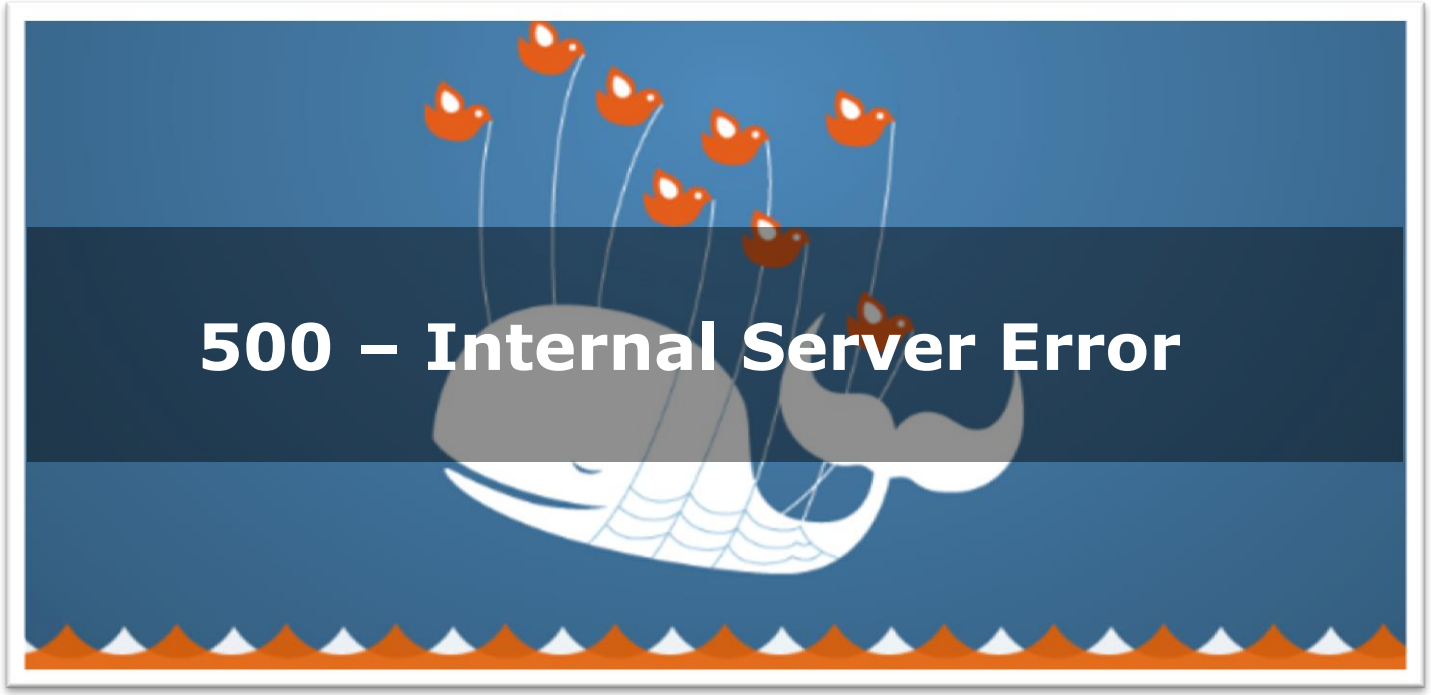
The "Digital Ready Architecture" requires a different approach and is based on a set of key assumptions/guidelines



SENACOR

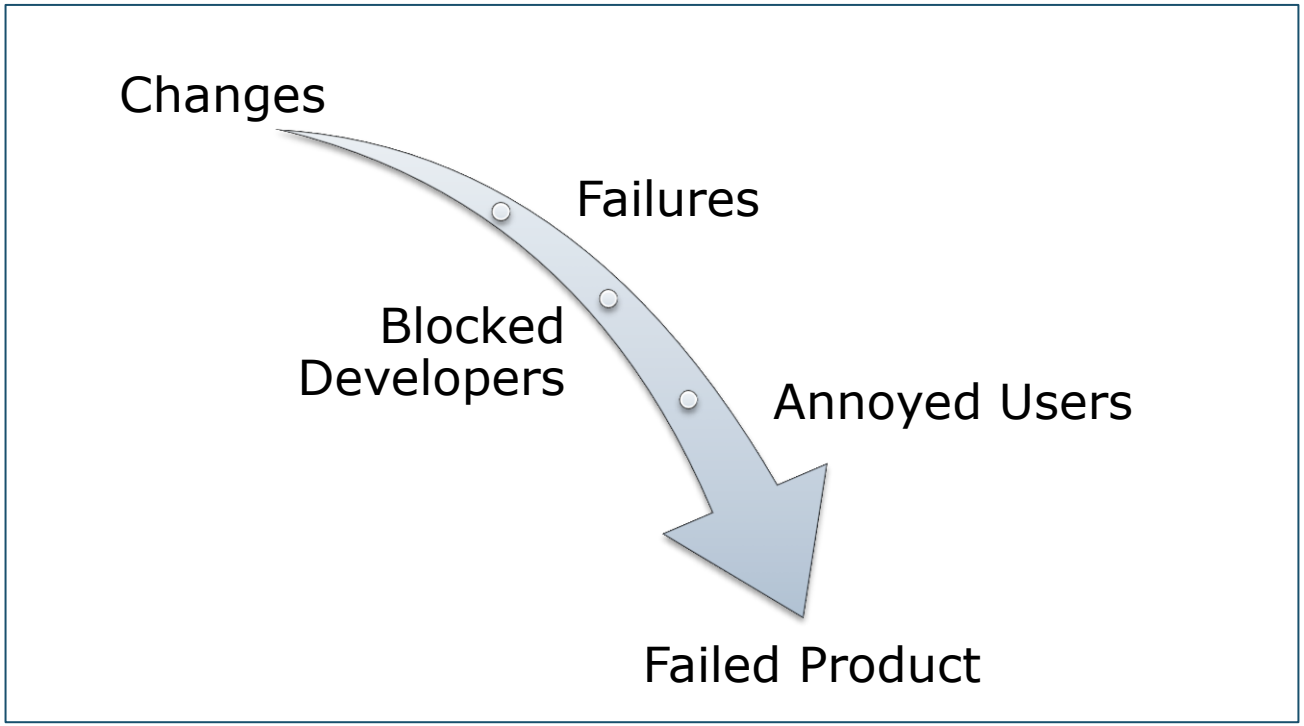
What happens when a service …

- **is down?**
- **is slow to respond?**
- **responds with an unexpected response?**
- **is thrashed with huge requests?**



SENACOR

Why to be resilient



- **Make many changes quickly,** hence be able to handle more failures
- **Safe costs,** free developers from emergency calls from production
- **Retain users,** avoid complete outages

SENACOR

How to be resilient

Resilience

The system stays responsive in the face of failure with a graceful degradation of service if required

Isolate failures by

- **Redundancy** (i.e. no single point of failure)
- **Loose coupling**
  - own data store
  - asynchronous messaging
  - location transparancy
  - eventual consistency

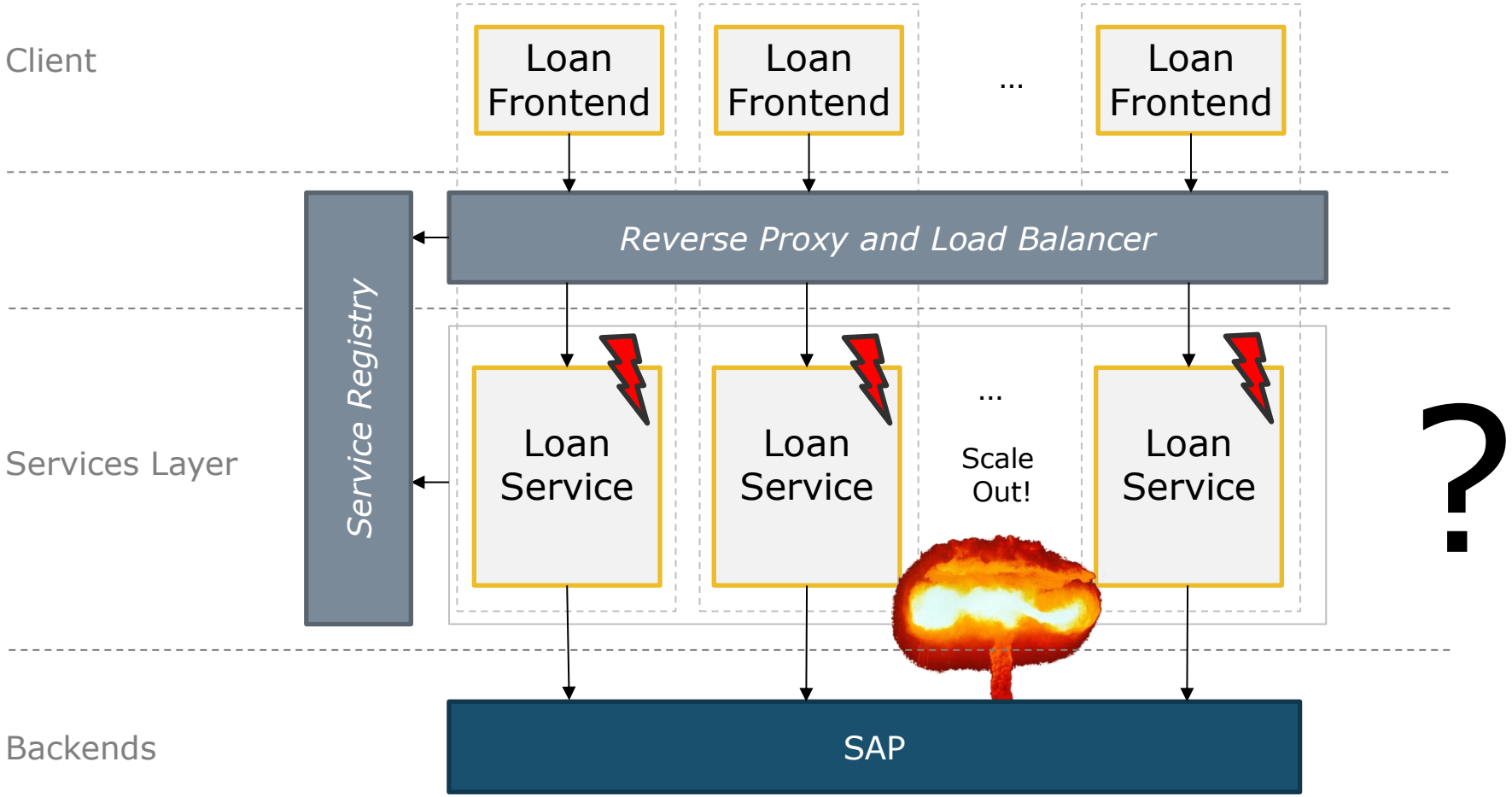
to be able to **fallback** on some default behavior i.e. cached data

Elasticity

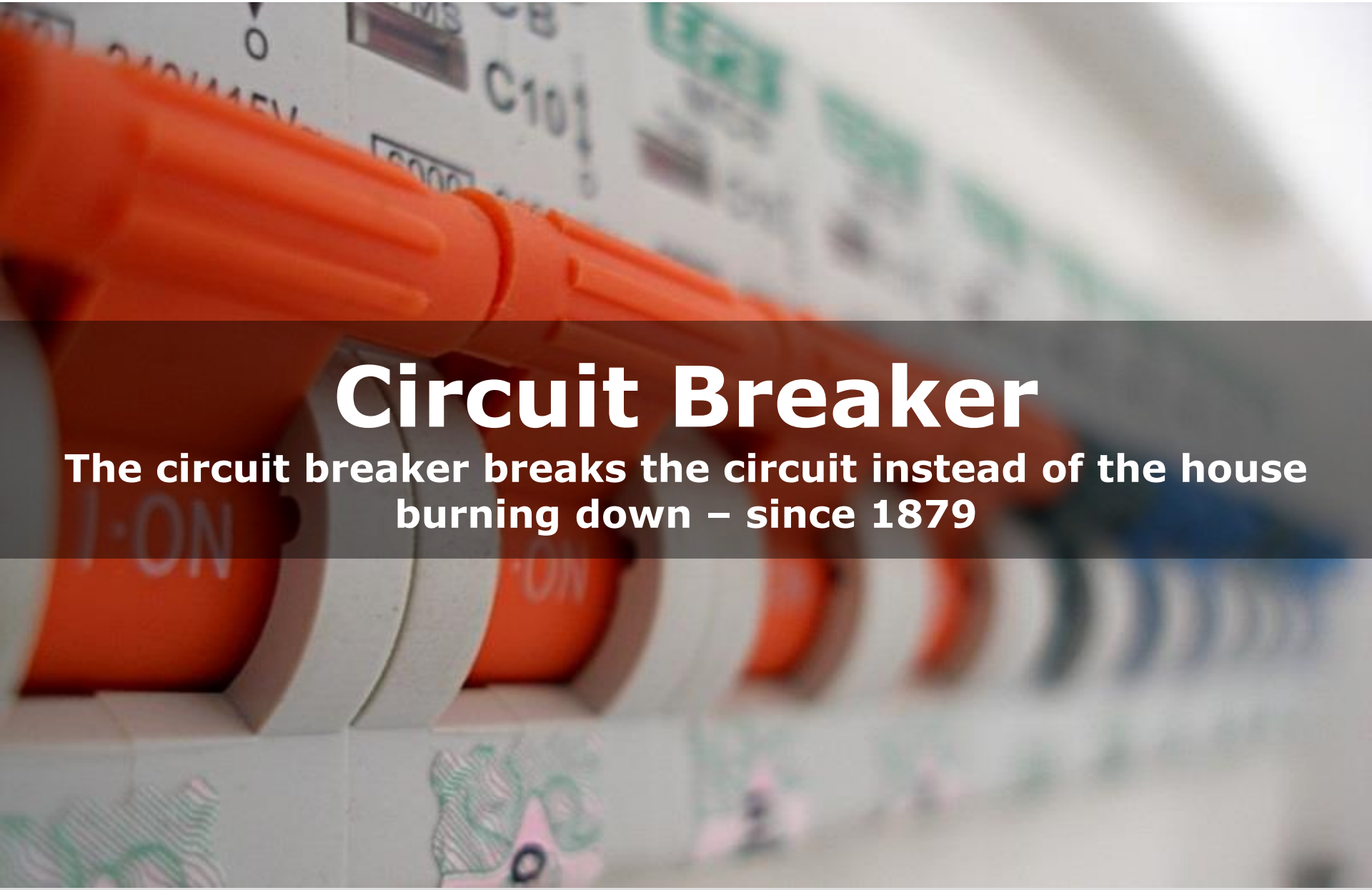
Reactivity

SENACOR

No System is an island …



SENACOR



Circuit Breaker

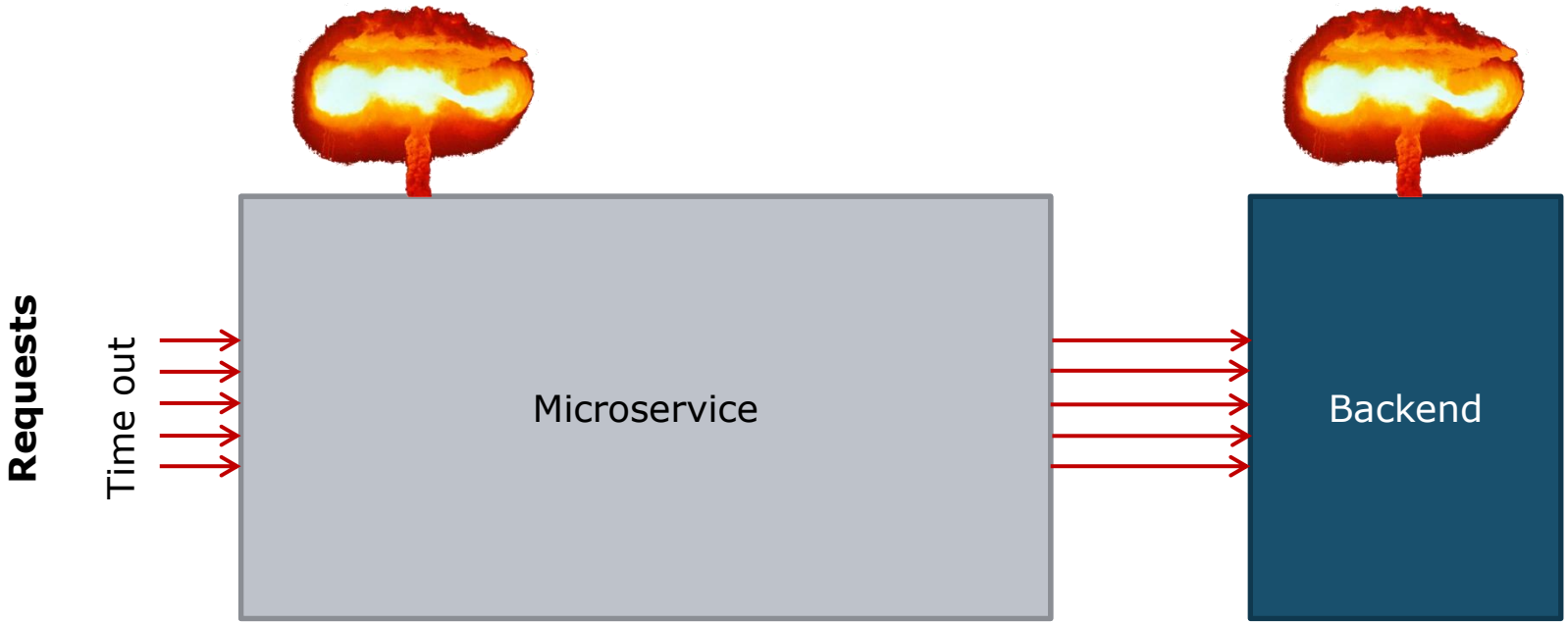
The circuit breaker breaks the circuit instead of the house burning down – since 1879

SENACOR

Unavailable services can affect the overall availability of synchronous, blocking systems

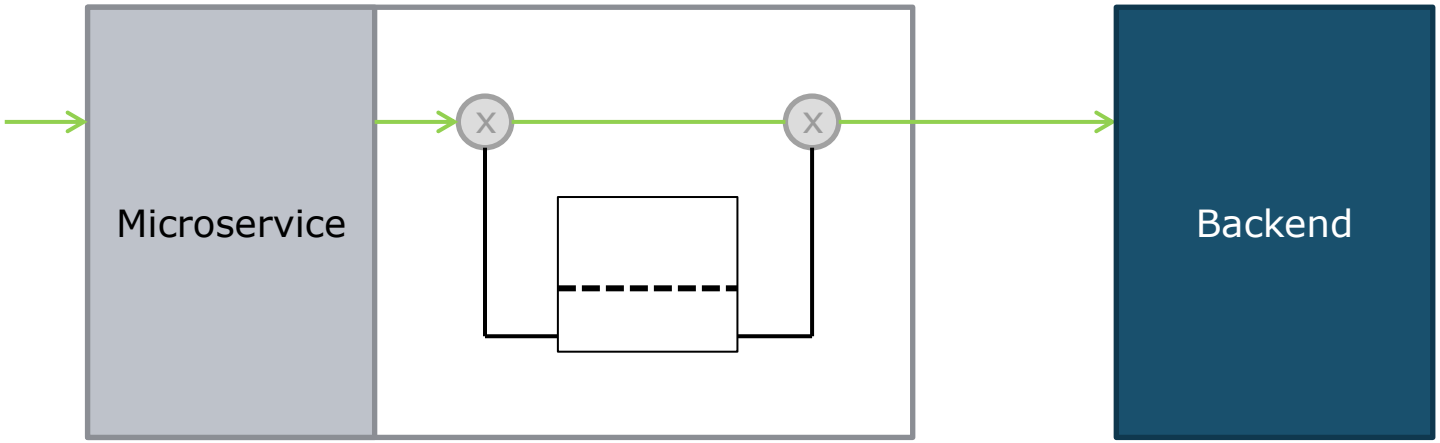
SENACOR

Unavailable services can affect the overall availability of synchronous, blocking systems



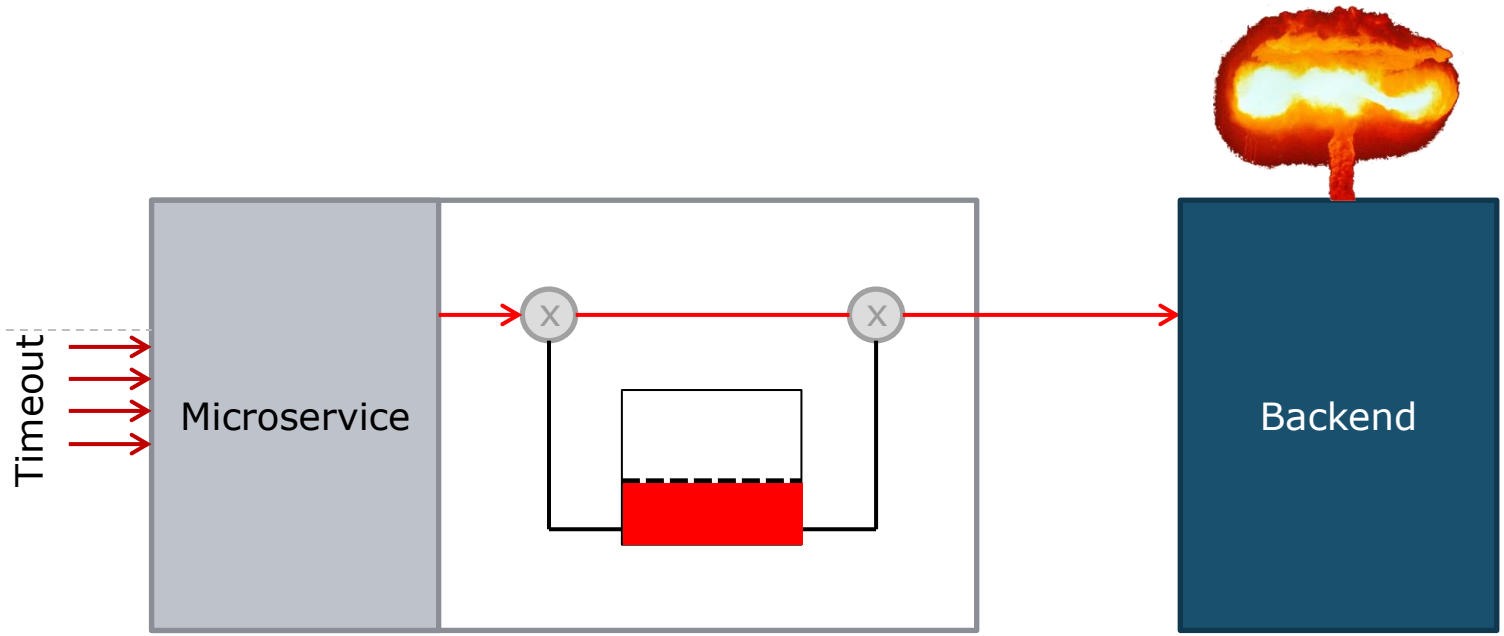
SENACOR

Circuit Breaker Pattern isolates backend access



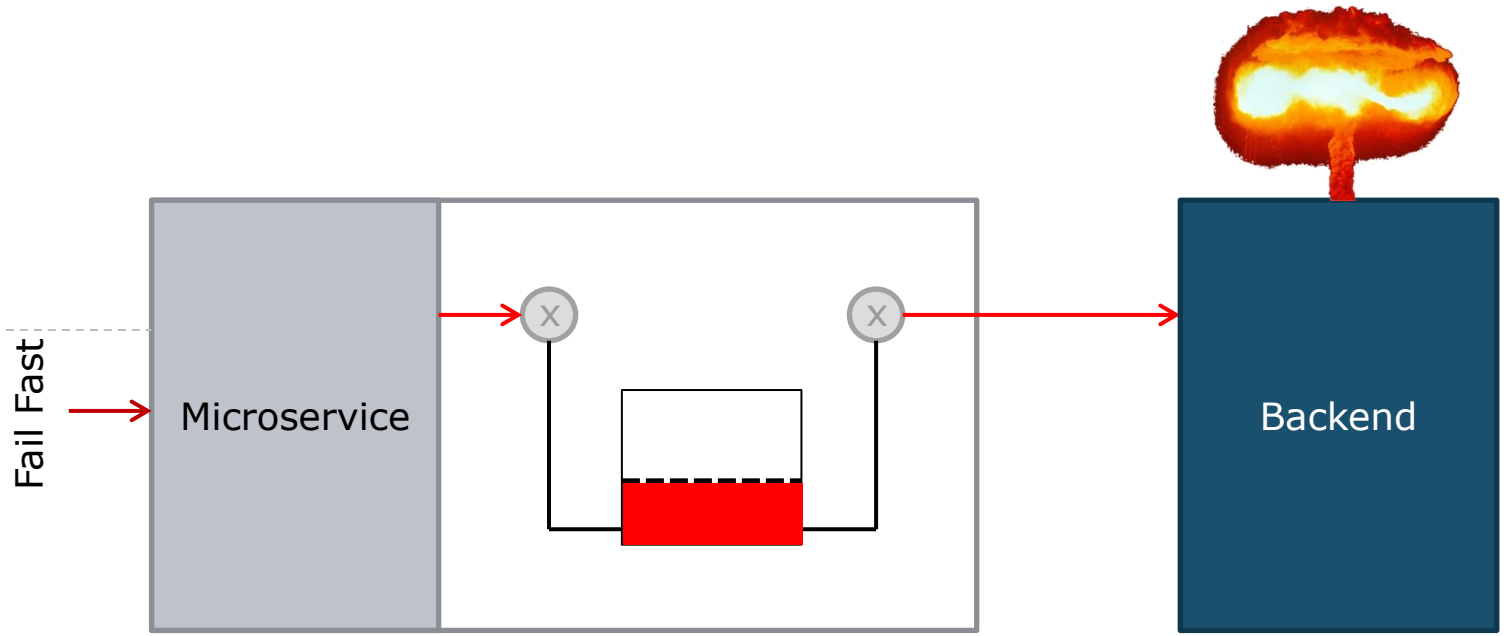
SENACOR

Circuit Breaker Pattern isolates backend access



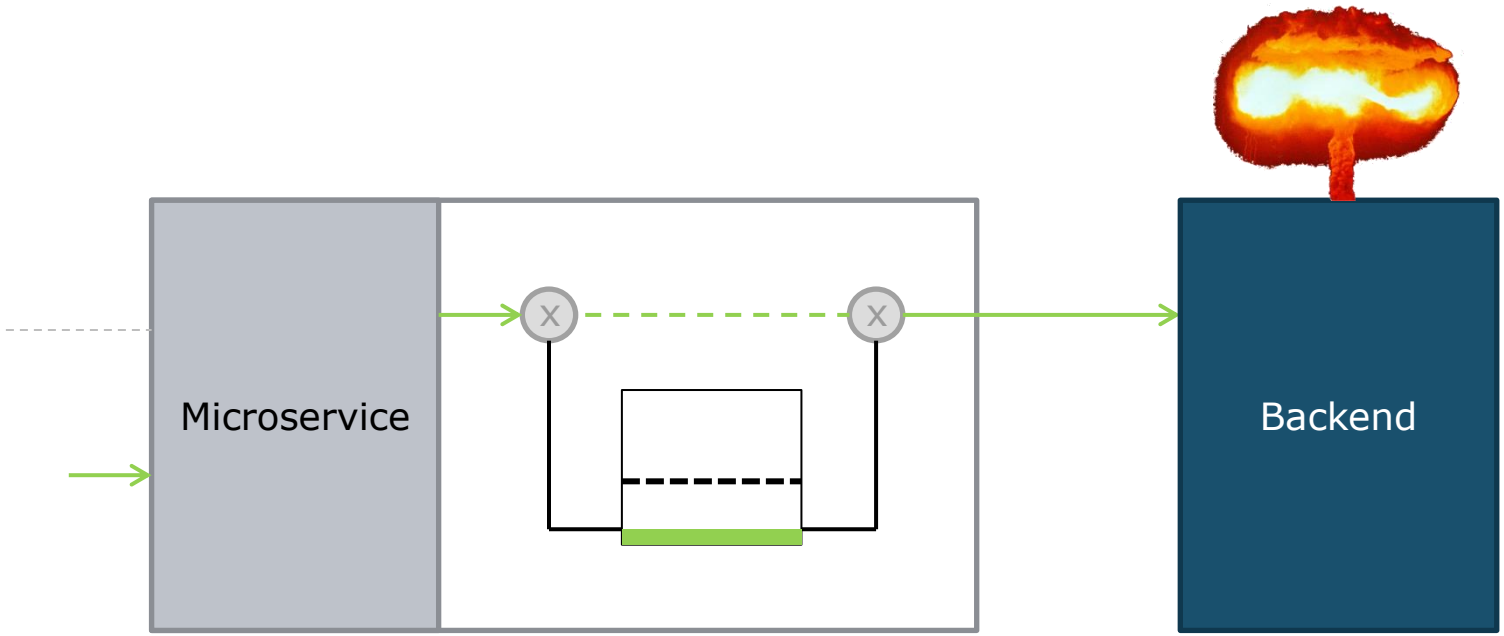
SENACOR

Circuit Breaker Pattern isolates backend access



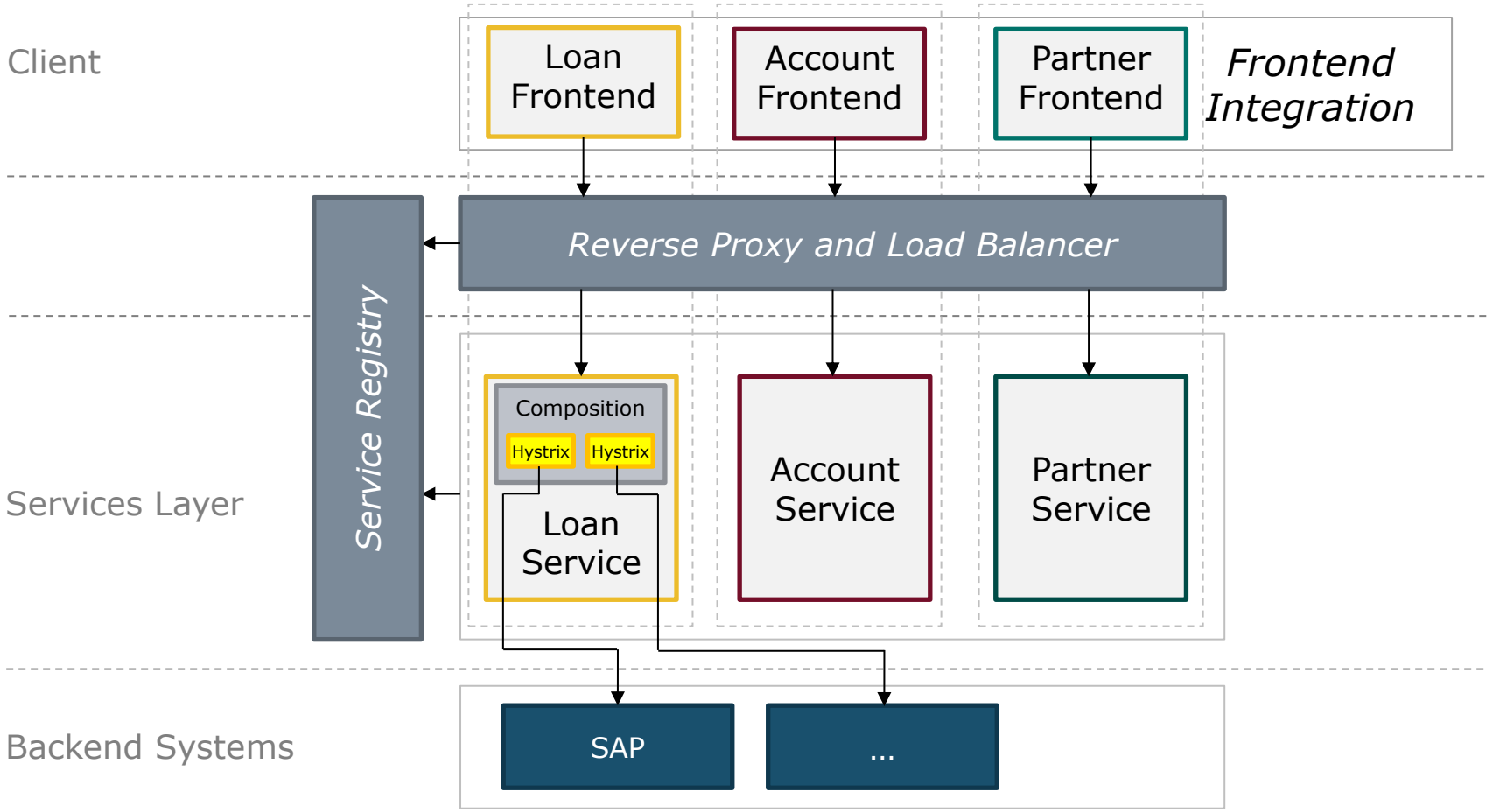
SENACOR

Circuit Breaker Pattern isolates backend access



SENACOR

Hystrix isolates invocations in Server-side service compositions



SENACOR

The "Digital Ready Architecture" requires a different approach and is based on a set of key assumptions/guidelines

SENACOR

Why being responsive

highly interactive (responsive) systems, which

- provide consistent response times
- are able to operate under high load and huge data volume

SENACOR

How to be responsive: Reactive frameworks are based on non-blocking, asynchronous, and event-driven implementations

✓ Non-Blocking API ✓ Asynchronicity ✓ Event-driven

We Are Reactive

VERT.X

RxNetty

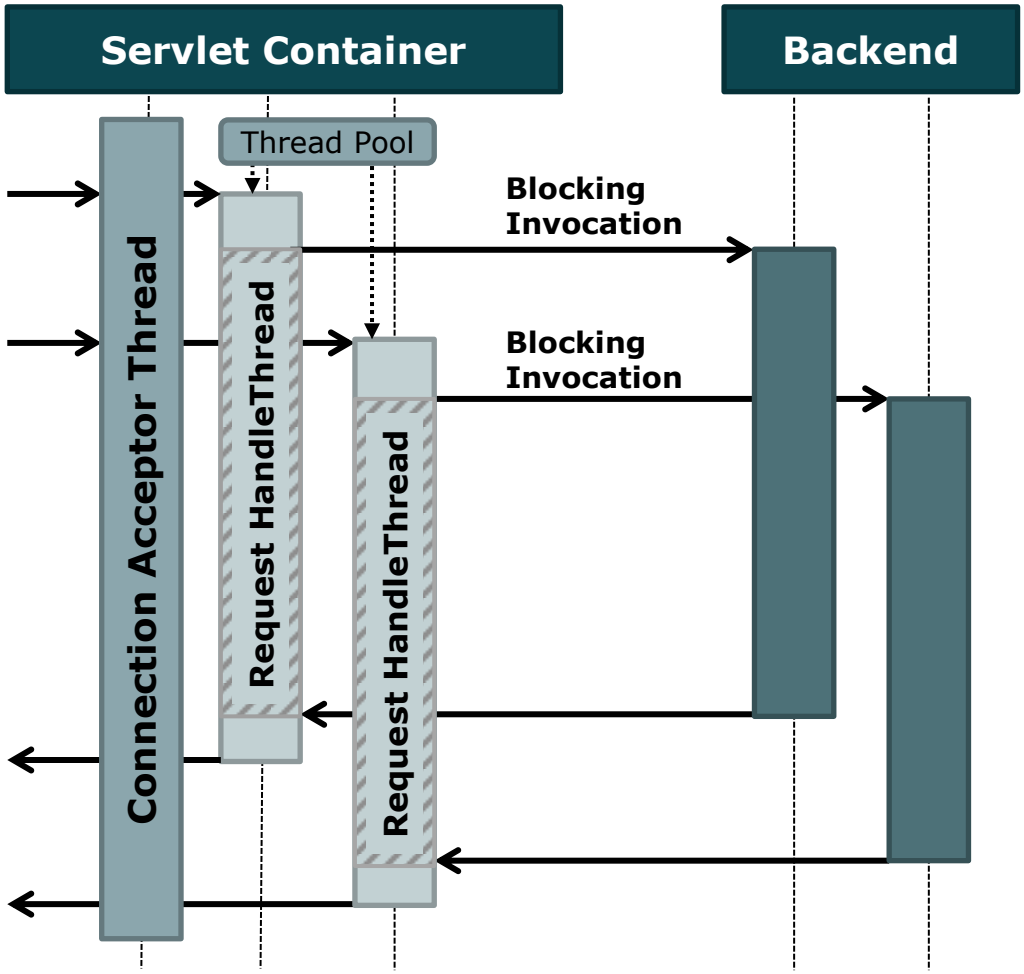
akka

node.js

spring

SENACOR

The servlet model uses one thread per request to facilitate an synchronous, blocking programming model

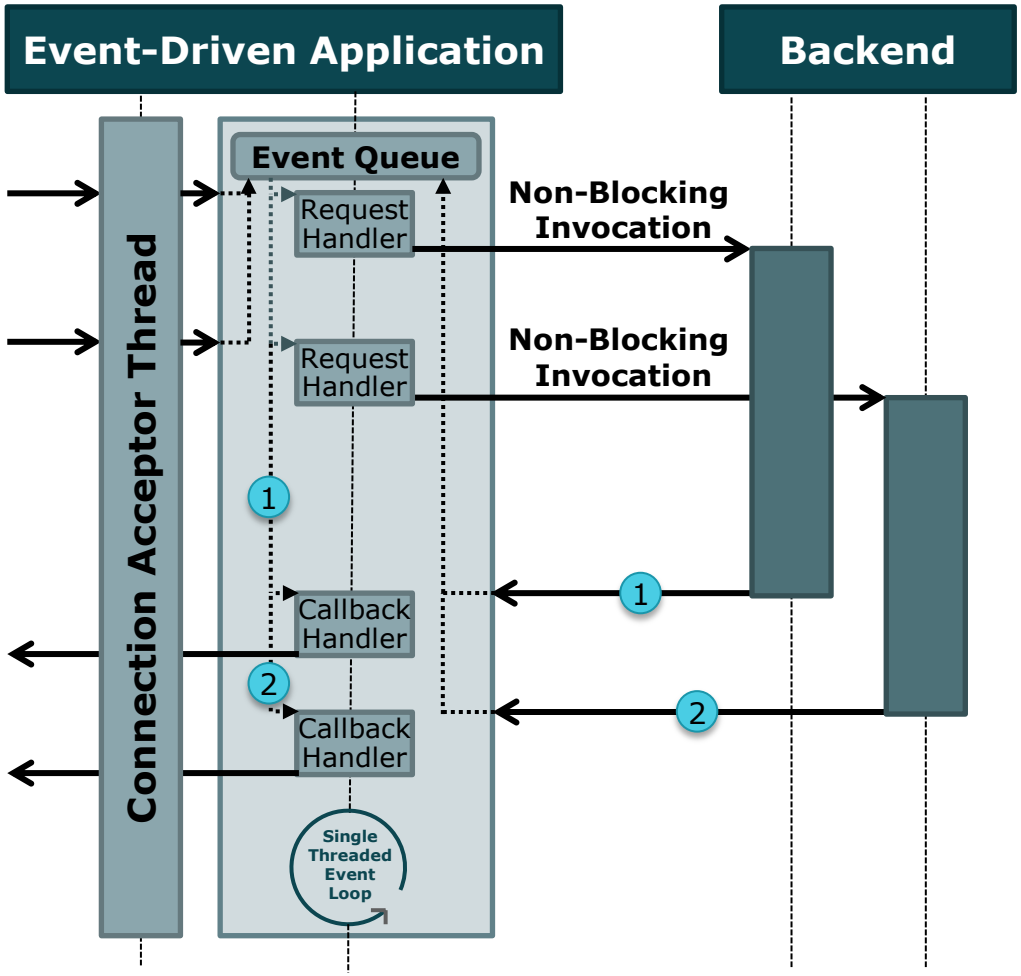


Servlet Model:
One thread per request

- Synchronous programming model
- Blocking Invocations
- Shared state and locks

SENACOR

Support a high number of parallel request using an asynchronous, non-blocking request handling



Event-driven application: Single-threaded event loop

- Asynchronous event handling
- No locks / synchronization required

SENACOR

Do you want to know more …

non-blocking API

Asynchronous

Event-driven

ReactiveX

Building reactive applications
with RxJava

Rx-Hackathon, 15:00

SENACOR

An architecture build on the principles of non-blocking, reactive Microservices meets the challenges of the digital age

	Evaluation
Embrace Change	▪ Microservices isolate changes and reduce the risk of releasing new business value ▪ Both technological and business innovations are furthered
Elasticity and Scalability	▪ Scalability is usually achieved by clustering and scaling vertically in advance ▪ Fine grained scalability is in generally hard to achieve
Embrace Failure	▪ Horizontal scalability enables elastic reaction to load ▪ Systems can be scaled on a fine grained level
Responsiveness and Reactivity	▪ Reactivity needs to be designed into the systems ▪ But: fine grained systems encourage the application of new techniques and methodologies

SENACOR

- Software is eating the world
- What is architecture?
- Status quo: Multi-Channel Architecture
- Digital-Ready Architecture
- **Assessment and conclusion**

SENACOR

…don't even consider Microservices unless you have a ***system that's too complex to manage as a monolith****.*
The ***majority*** *of software systems should be built as* ***a single monolithic application****.*

- Martin Fowler

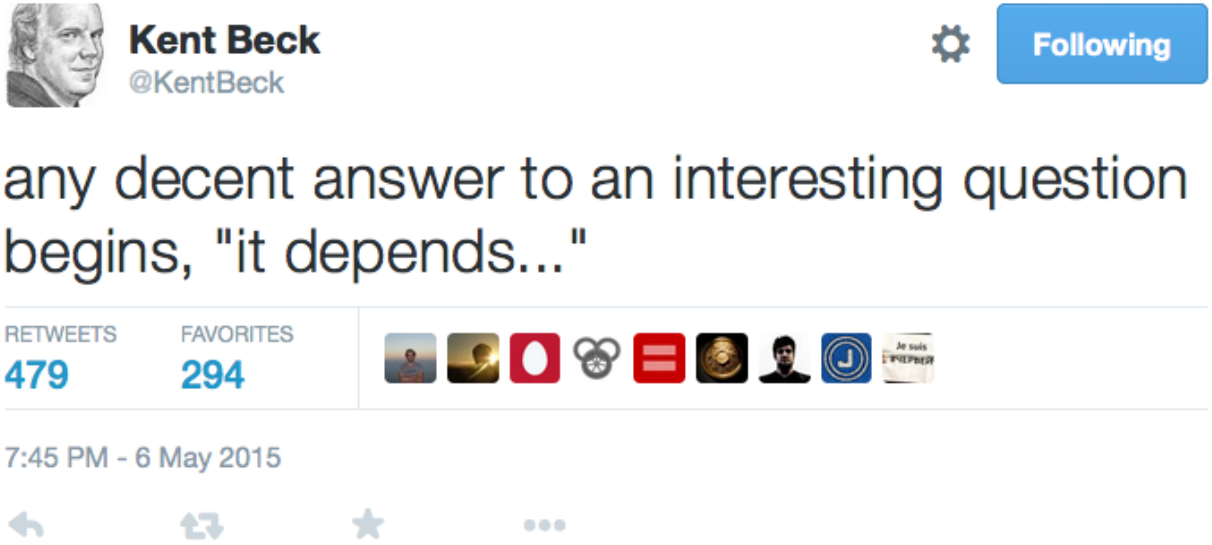
http://martinfowler.com/bliki/MicroservicePremium.html

SENACOR

If you decide to build things using a microservices approach, …while it will be a lot easier to make ***localized decisions*** *in each individual part, it will be much* ***harder to change the very boundaries that enable this****.*

- Stefan Tilkov

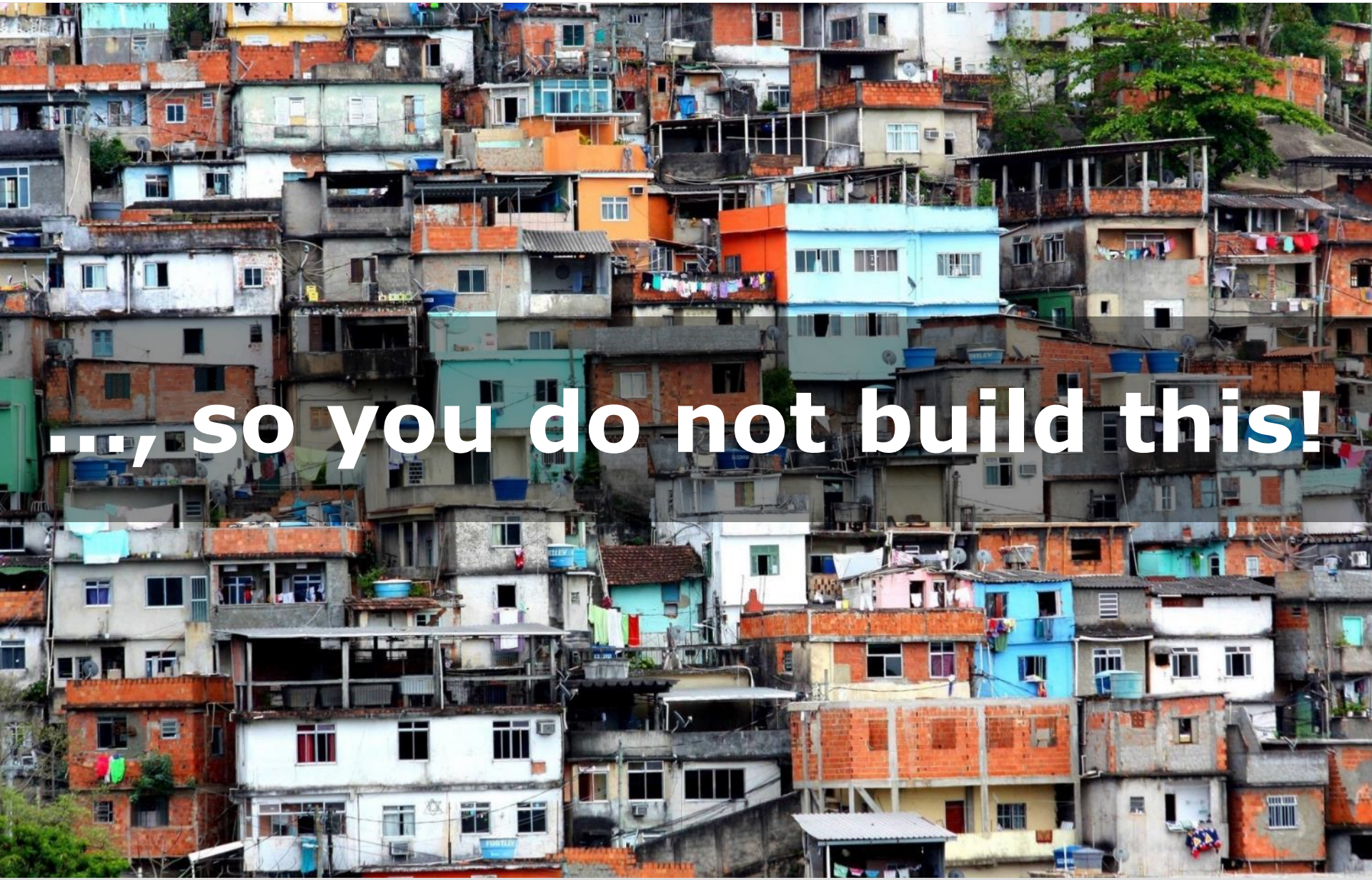
http://martinfowler.com/articles/dont-start-monolith.html


Kent Beck
@KentBeck
Following
any decent answer to an interesting question begins, "it depends..."
RETWEETS
479
FAVORITES
294
7:45 PM - 6 May 2015

SENACOR

Think before your act,…

SENACOR



…, so you do not build this!

SENACOR

The age of digitization acts as a disruptive force and requires an end-to-end rethinking of established principles and patterns

Software is eating the World

- Disrupting forces put pressure on the IT departments of corporations
- In order to meet the market demands, new ideas and products must be delivered faster and more often
- Bank of 2 Speeds – think speedboats and tankers

Monolithic apps. are not up to the challenge

- Systems are build according to the organization's communication structure
- Traditional software system architectures, such as the multi-channel architecture, make meeting market demands challenging

Self-Contained-Systems offer a solution

- Going from monolithic applications to as system of systems requires change on an organizational level
- You have to understand your domain
- Going from monolithic applications to as system of systems requires change on an organizational level

Questions? Comments? Feedback?

Dr. Michael Menzel
Architect

michael.menzel@senacor.com
www.senacor.com

Daniel Heinrich
Senior Developer

daniel.heinrich@senacor.com
www.senacor.com

SENACOR

Images links / Copyright belongs to resp. author

- Rick Ligthelm ligthelm@flickr
- Ben Babcock tachyondecay@flickr
- PhotoArt Laatzen trombone65@flickr
- Leslie Lamport wikipedia.org
- Steve Watts quicksandala@morguefile.com
- DAVINCI Haus 62694216@N05@flickr
- Susanne Nilsson infomastern@flickr
- Kurtis Garbutt kjgarbutt@flickr
- Oliver Widder geek-and-poke.com
- Angus MacRae gustaffo89@flickr
- Lynne Cazaly lynnecazaly.com.au
- ctbto@flickr
- Stefano Leotta mylife1@flicker
- Osvaldo Gago fotografar@flickr
- Moyan Brenn aigle_dore@flickr
- Martin Fowler martinfowler.com
- leafbug@flickr
- Sonya Ong and Travis Hydzik SonyaandTravis.com
- Tony Guyton tonzpalmer24@flickr
- Timothy Pearce timpearcelosgatos@flickr
- dany13@flickr