



Una visión moderna de las arquitecturas

soprasosteria

commit
MAD · NOV 23-24 · 2018



José Manuel López Doña

CTO (Spain) at Sopra Steria

<https://www.linkedin.com/in/jmlopezdona>

@jmlopezdona



José Alberto Ruiz Casarrubios

Head of Modern Architecture at Sopra Steria

<http://jaruiz.io>

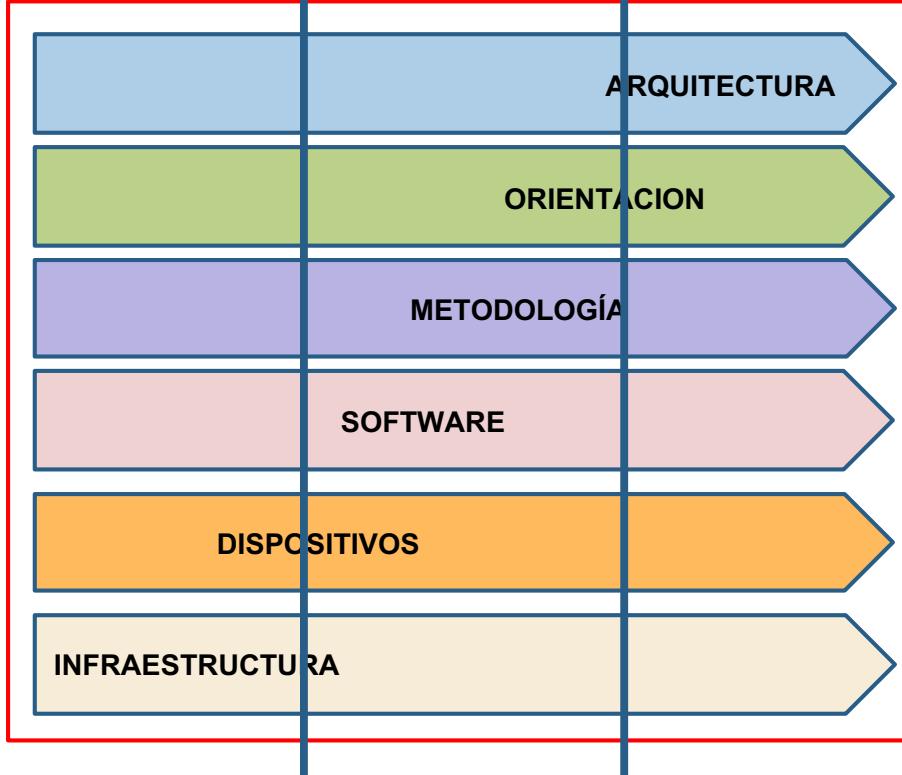
<https://www.linkedin.com/in/jaruizcasarrubios>

@jalg80

sopra  steria

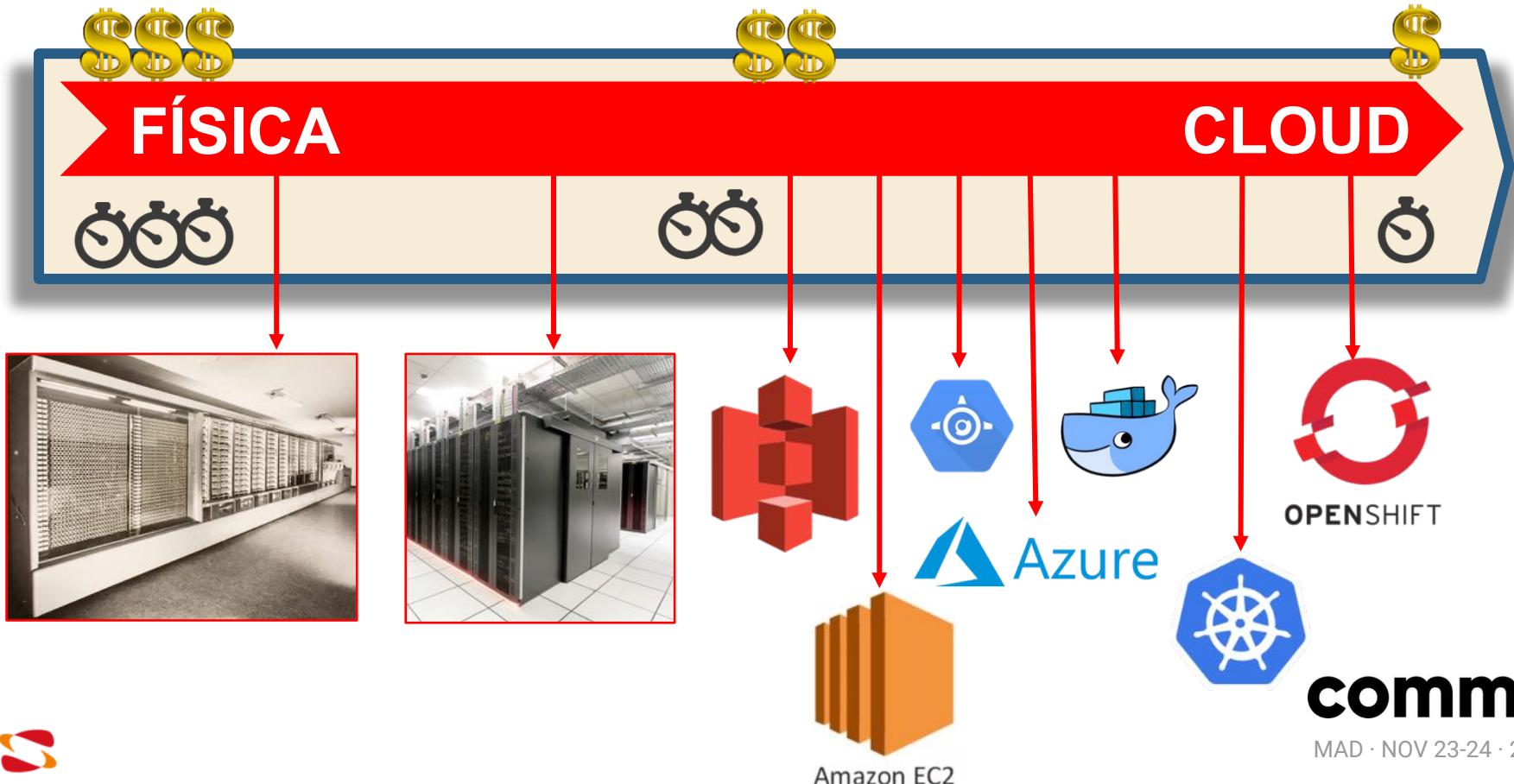
commit

MAD · NOV 23-24 · 2018



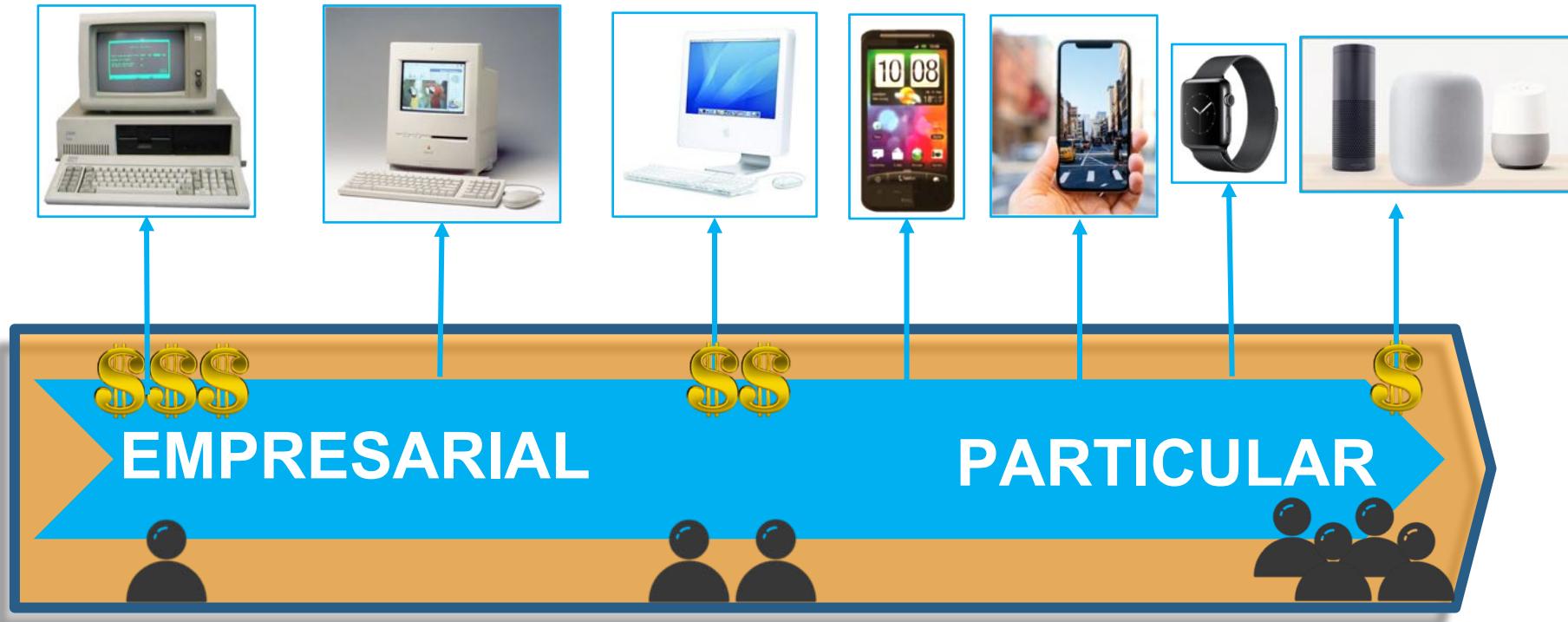
Un poco de contexto

INFRAESTRUCTURA



commit

MAD · NOV 23-24 · 2018



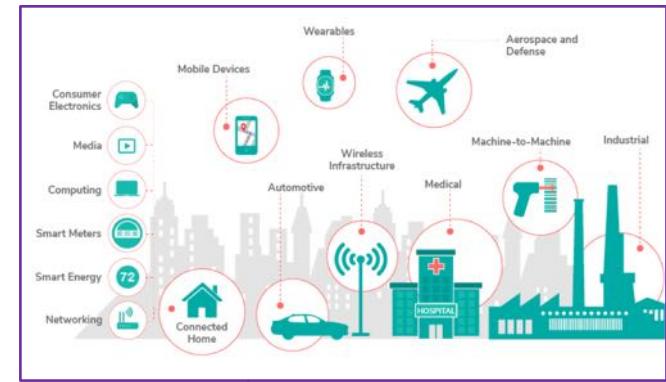
DISPOSITIVOS / CANALES



commit

MAD · NOV 23-24 · 2018

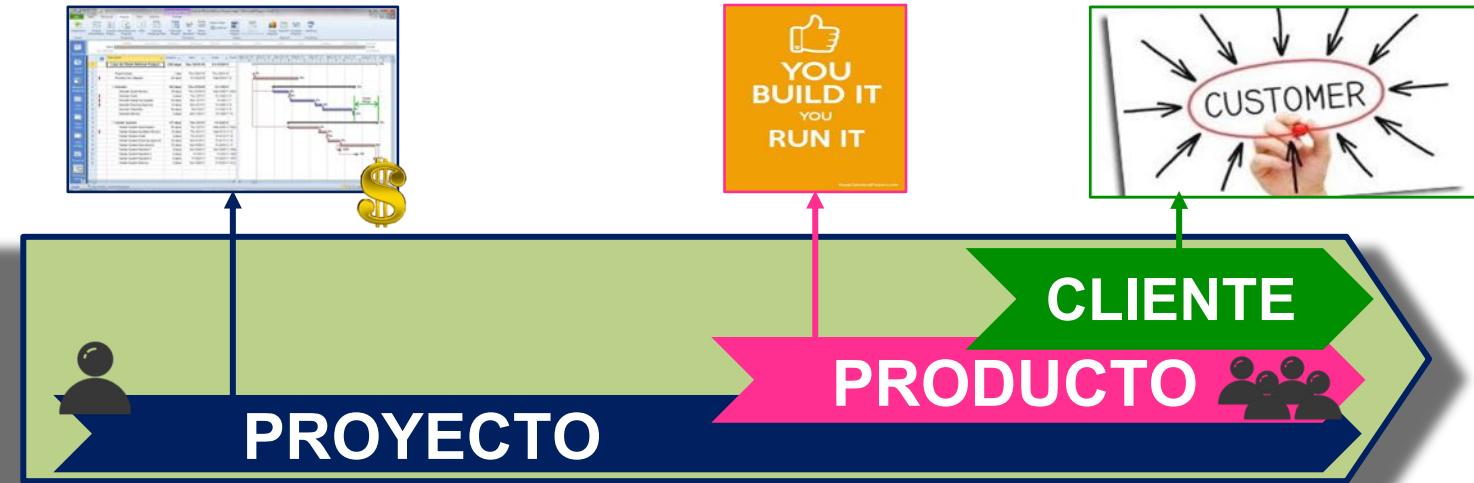
SOFTWARE



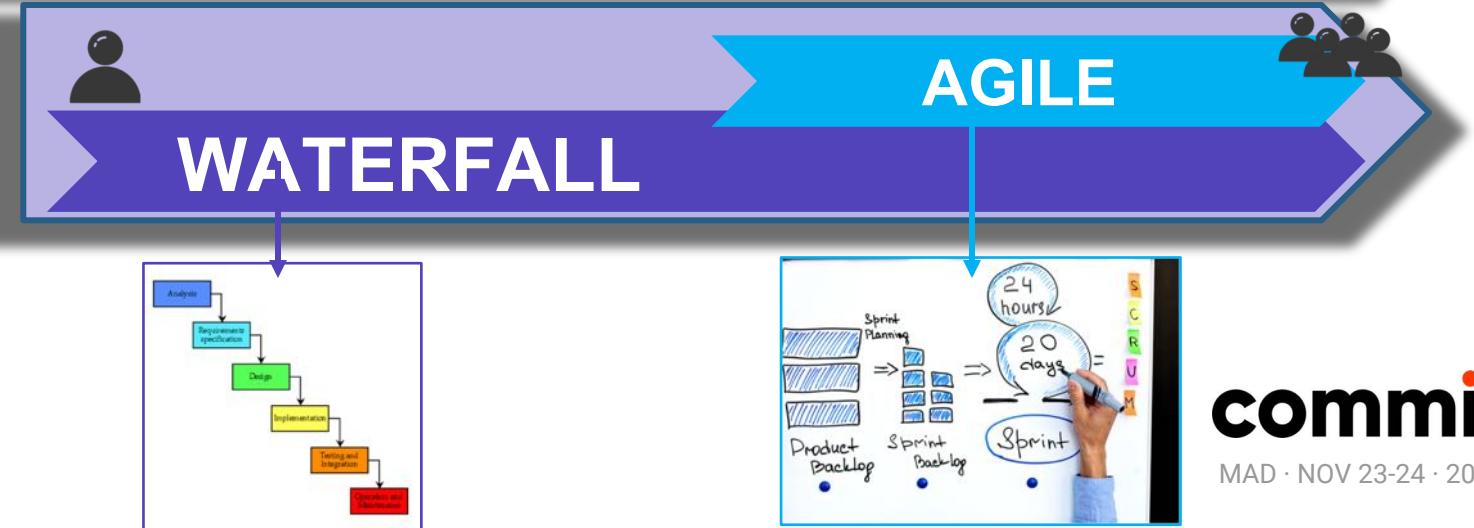
commit

MAD · NOV 23-24 · 2018

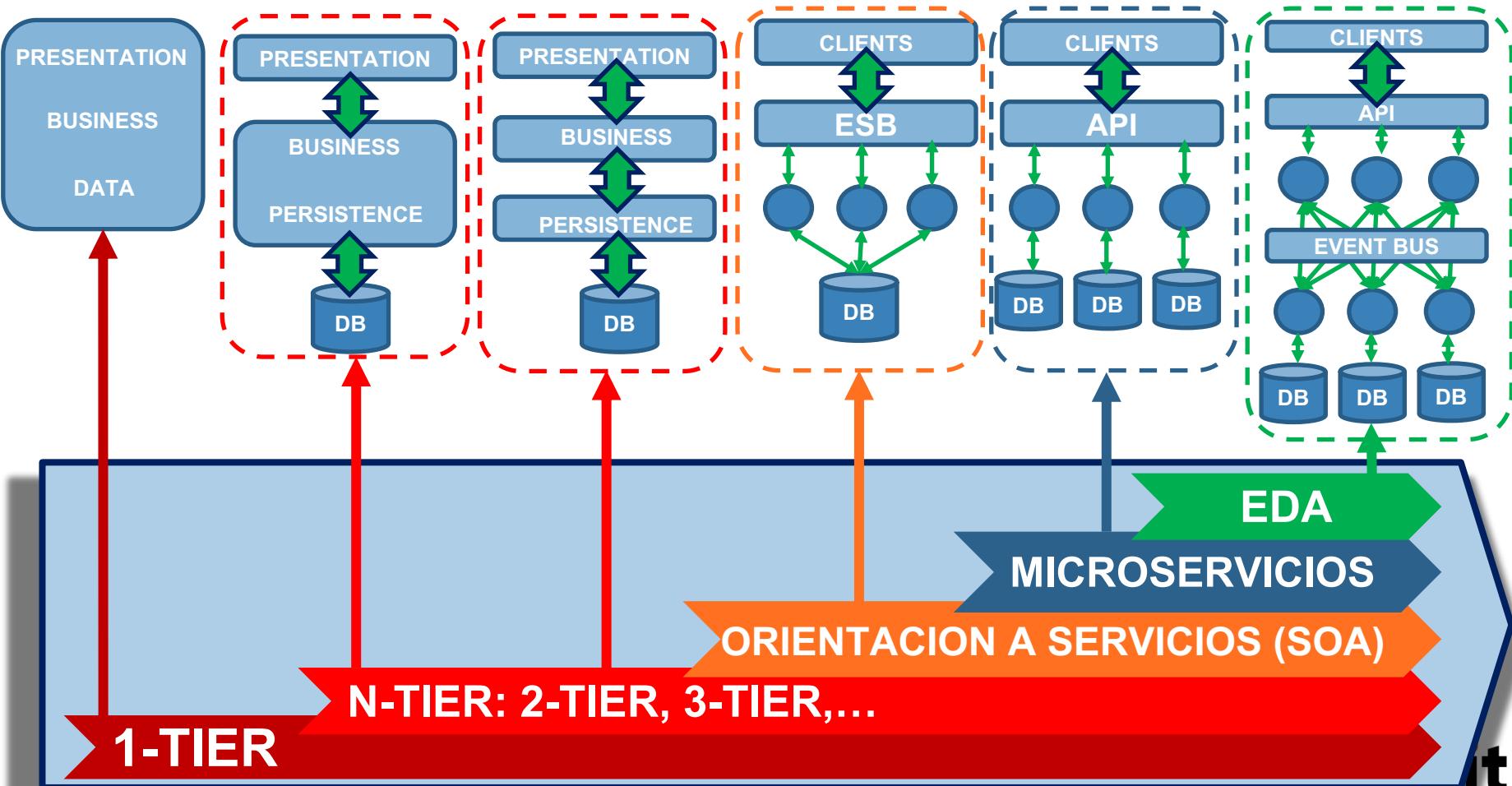
ORIENTACION



METODOLOGÍA



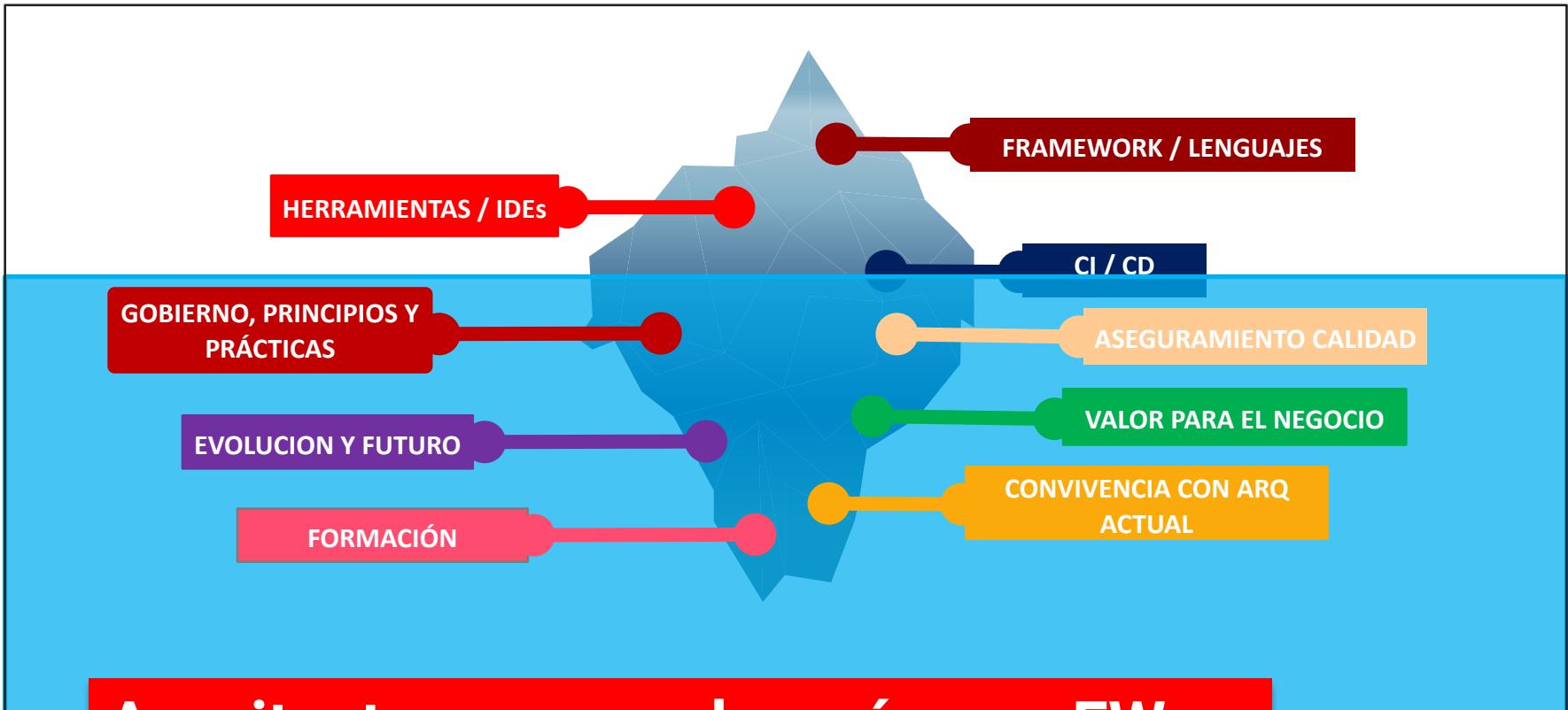
commit
MAD · NOV 23-24 · 2018



ARQUITECTURA



¿Qué entendemos
nosotros por
Arquitectura?

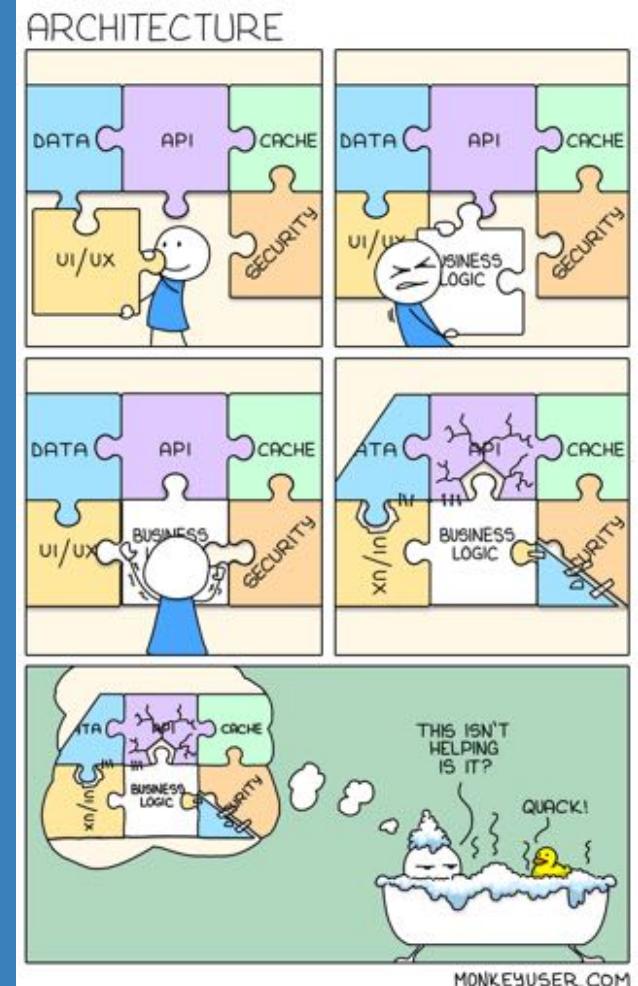


Arquitectura es mucho más que FWs o herramientas



ARQUITECTURA

*“Conjunto de prácticas, principios, técnicas, herramientas o tecnologías sobre las que se construyen las aplicaciones y servicios para **satisfacer los requerimientos de Negocio** planteados en la organización teniendo en cuenta su **contexto**”*



¿Cuáles son tus NFR o "-ilities"?

The Architecture "-ilities"

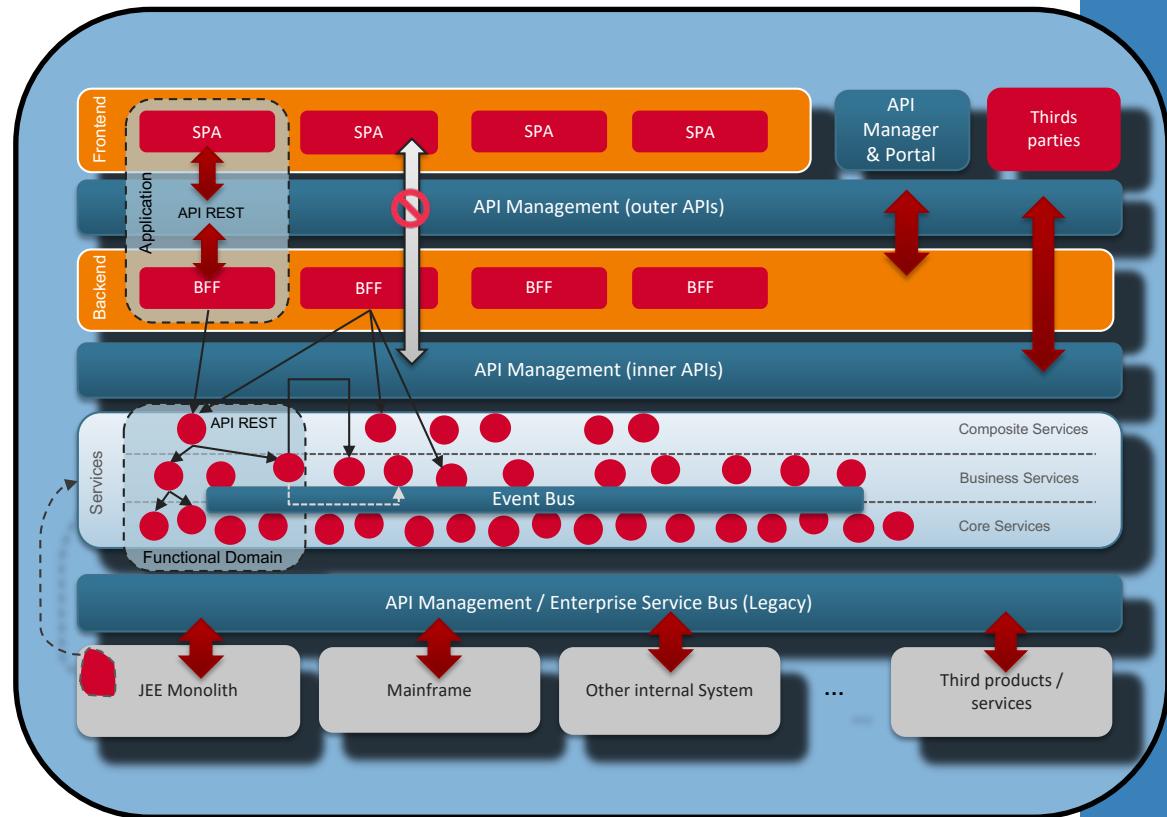
accessibility	deployability	manageability	reusability
accountability	discoverability	modifiability	robustness
accuracy	distributability	modularity	safety
adaptability	durability	operate	scalability
administrability	effectiveness	ability	seamlessness
affordability	efficiency	orthogonality	self-sustainability
auditability	usability	portability	serviceability
autonomy	extensibility	precision	secure
availability	failure-transparent	predictability	ability
compatibility	ency	producibility	simplicity
composability	fault-tolerance	probability	stability
configurability	fidelity	recoverability	standards-compliance
correctness	flexibility	relevance	survivability
credibility	inspectability	reliability	sustainability
customizability	installability	repeatability	tailorability
debugability	Integrity	reproducibility	testability
degradability	interoperability	resilience	timeliness
determinability	learnability	responsiveness	traceability
dependability	maintainability		



commit

MAD · NOV 23-24 · 2018

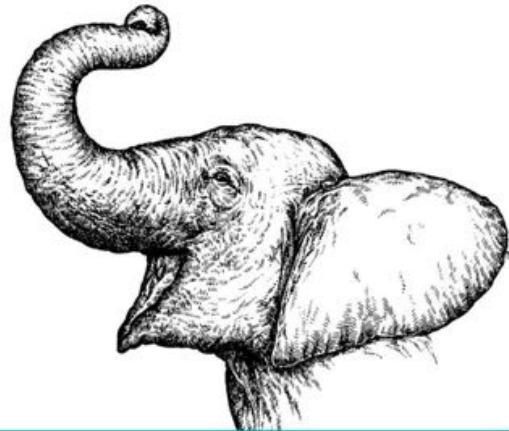
¿Crees que es una buena Arquitectura?



“La mejor arquitectura es aquella que está adaptada en cada momento a las necesidades de Negocio de tu organización y no otra”



The answer to every programming question ever conceived



It Depends

The Definitive Guide

O RLY?

@ThePracticalDev

commit
MAD · NOV 23-24 · 2018

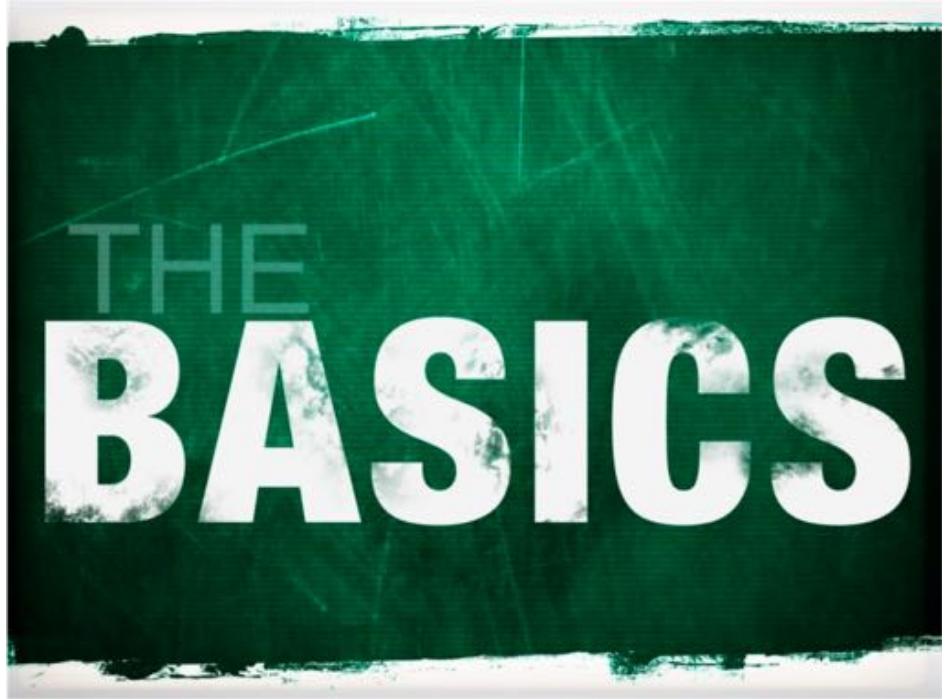


Tú Negocio cambia,
evoluciona, se adapta o
muere

Tú Arquitectura debe
cambiar, evolucionar,
adaptarse a las nuevas
necesidades de Negocio
o muere



¿Hablamos de Microservicios?

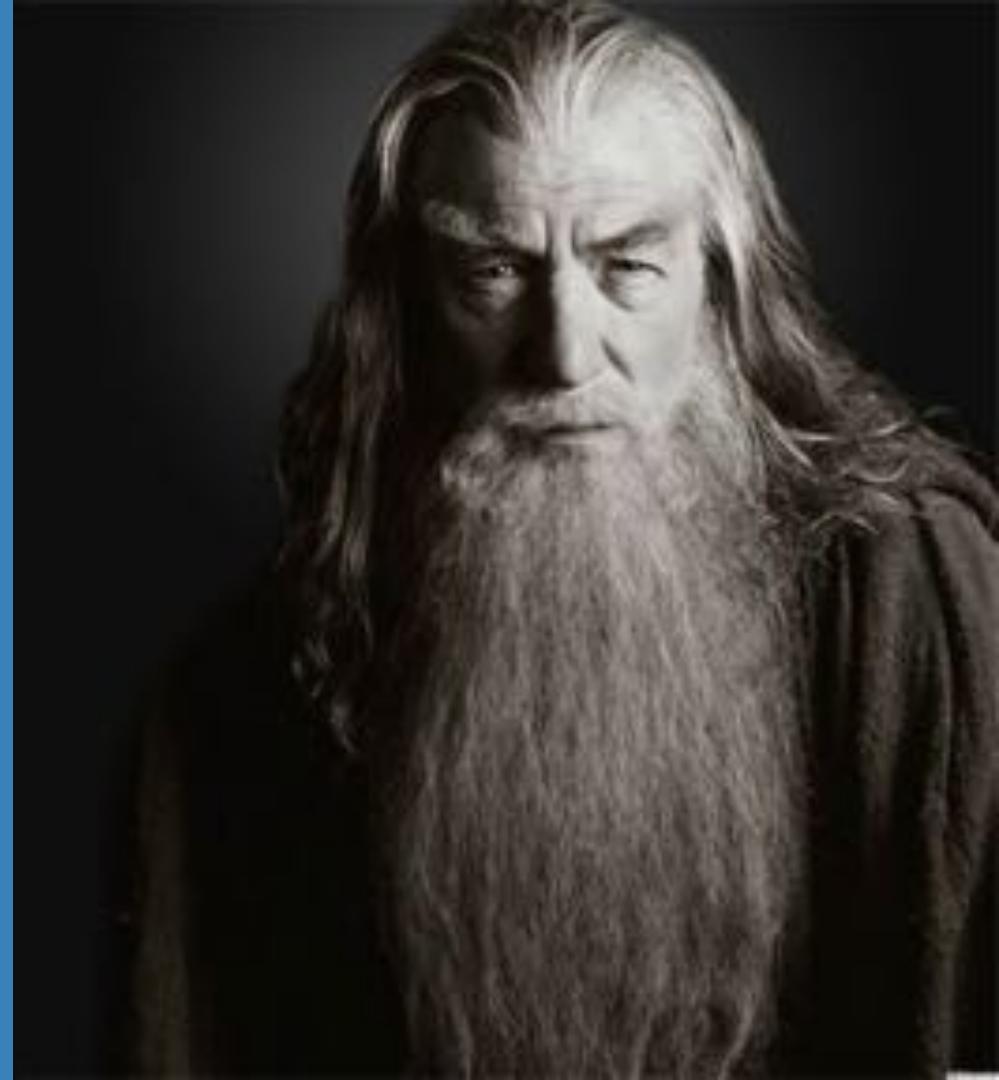


commit
MAD · NOV 23-24 · 2018

Microservicios

no es Spring Boot, Docker
o Kubernetes.

Es un estilo de arquitectura



Micro-services, often deployed out-of-container or using an embedded HTTP server, are a move away from traditional large technical services. This approach trades benefits such as maintainability for additional operational complexity. These drawbacks are typically addressed using infrastructure automation and continuous deployment techniques. On balance, micro-services are an effective way of managing technical debt and handling different scaling characteristics especially when deployed in a service oriented architecture built around business capabilities.

A decade ago when memory was at a premium, application servers made a lot of sense. They were popular and useful as a mechanism to run and manage multiple applications on a shared server or cluster. These days applications are more often run on separate physical or virtual servers and the need for an application server is reduced. Consider evaluating **server / application container end-of-life** within your organization, and only use one if you benefit from the added complexity.

Techniques

— Adopt —

- 1. Automate database deployment
- 2. Coding architects
- 3. Continuous Delivery (CD)
- 4. Data visualizations of development and operations
- 5. DevOps
- 6. Emergent design
- 7. Evolutionary architecture

- 8. Evolutionary database
- 9. Health check pages
- 10. Simple performance trending
- 11. Test at the appropriate level

— Trial —

- 12. Acceptance test of journeys
- ▲ 13. Agile analytics
- 14. Build your own radar
- 15. Categorization & prioritization of technical debt
- ▲ 16. Embed
- 17. Event
- ▲ 18. Infrastructure development
- ▲ 19. Out-of-the-box
- 20. Performance testing a first

— Assess —

- ▲ 21. Single command deploy
- 22. Thoughtful caching
- ▲ 23. Windows infrastructure automation
- ▲ 24. Event sourcing
- ▲ 25. Experience Design (XD)
- ▲ 26. Mechanical sympathy
- ▲ 27. Micro-services
- ▲ 28. Production immune system

— Hold —

- 29. Database based integration
- 30. Feature branching
- 31. Manual infrastructure management
- 32. Scrum certification
- ▲ 33. Server / application container end-of-life
- ▲ 34. Test recorders

Technology Radar

<http://www.thoughtworks.com/radar>

Prepared by the ThoughtWorks
Technology Advisory Board

MARCH 2012

Microservices

a definition of this new architectural term

The term "Microservice Architecture" has sprung up over the last few years to describe a particular way of designing software applications as suites of independently deployable services. While there is no precise definition of this architectural style, there are certain common characteristics around organization around business capability, automated deployment, intelligence in the endpoints, and decentralized control of languages and data.

25 March 2014



James Lewis

James Lewis is a Principal Consultant at ThoughtWorks and member of the Technology Advisory

Board. James' interest in building applications out of small collaborating services stems from a background in integrating enterprise systems at scale. He's built a number of systems using microservices and has been an active participant in the growing community for a couple of years.



Martin Fowler

Martin Fowler is an author, speaker, and general loud-mouth on software development. He's long been puzzled by the problem of how to componentize software systems, having heard more vague claims than he's happy with. He hopes that microservices will live up to the early promise its advocates have found.

Translations: Japanese - Russian -

<https://martinfowler.com/articles/microservices.html>

Contents

- Characteristics of a Microservice Architecture
 - Componentization via Services
 - Organized around Business Capabilities
 - Products not Projects
 - Smart endpoints and dumb pipes
 - Decentralized Governance
 - Decentralized Data Management
 - Infrastructure Automation
 - Design for failure
 - Evolutionary Design
- Are Microservices the Future?

Sidebars

- How big is a microservice?
- Microservices and SOA
- Many languages, many options
- Battle-tested standards and enforced standards
- Make it easy to do the right thing
- The circuit breaker and production ready code
- Synchronous calls considered harmful



Principios básicos

Componentes
Como
Servicios

Organizados
alrededor
del negocio /
equipos
cross-func

Orientación a
producto,
No proyecto

“Smart endpoints
Dump pipes”
(no ESBs / BPELs)

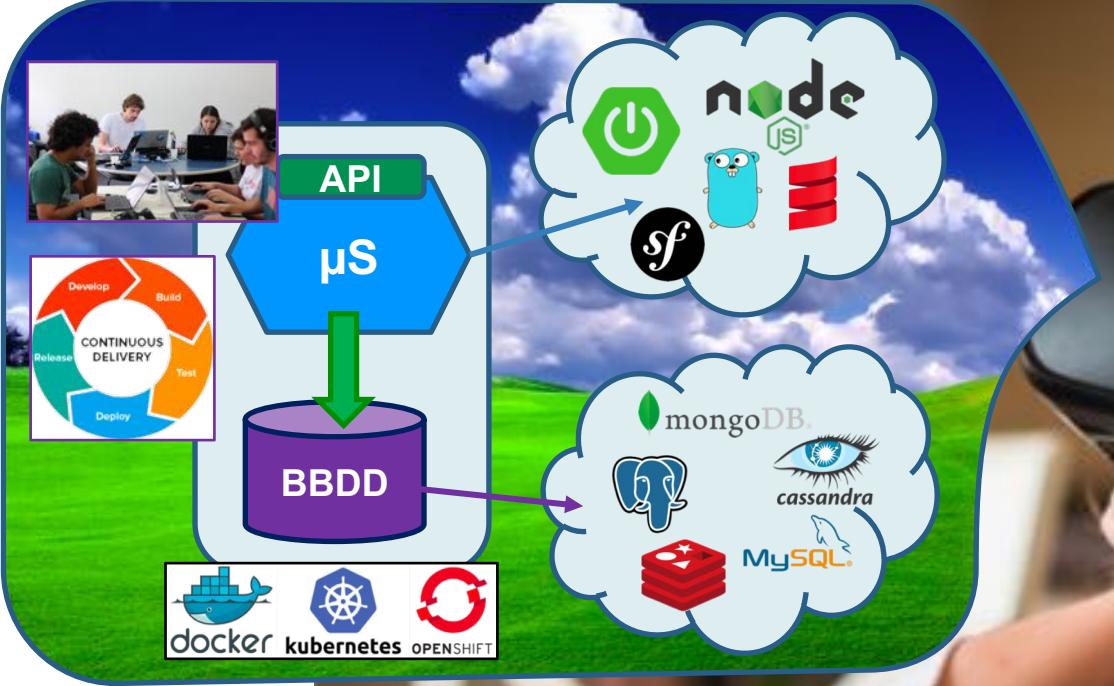
Gobierno
descentralizado
(Poliglotismo)

Gestión
descentralizada de
datos
(DDD, BBDD
Propia)

Diseño para
el fallo

Automatización
y
entrega
continua

Diseño evolutivo
y
para evolucionar



Este es tu
“greenfield”



commit
MAD · NOV 23-24 · 2018



Y unos meses
después será
algo así...

¿Te suena?

Construir Microservicios es “relativamente” sencillo

Construir una
Plataforma de Microservicios
para dar soporte a tu Negocio,
no es tan sencillo

Reutilización

Transacciones

Agregación de
datos

Event
Sourcing

Aplicaciones

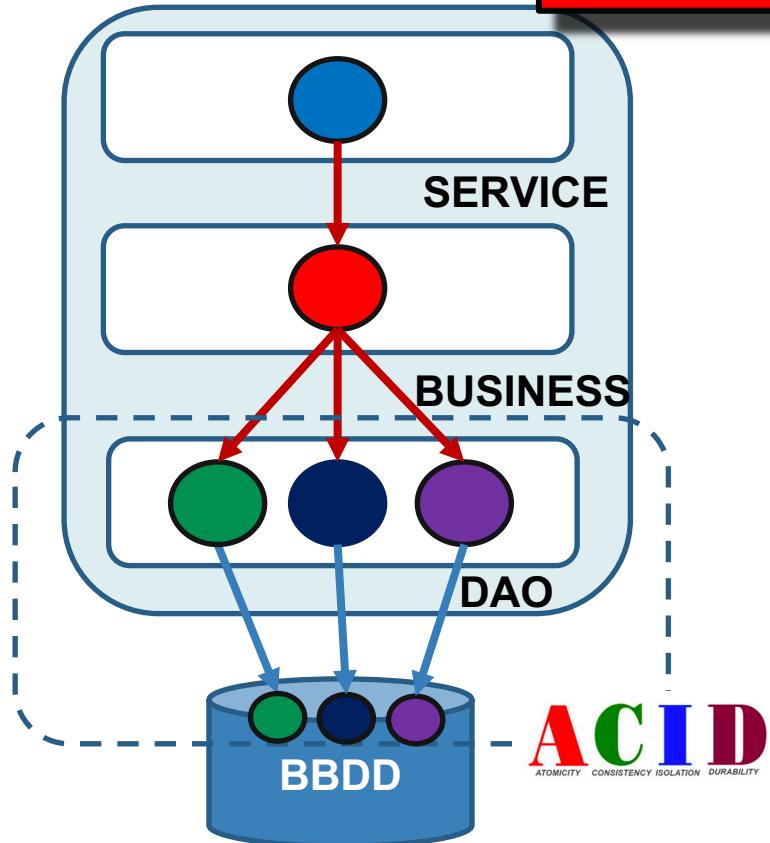


A photograph of a woman with blonde hair singing into a microphone. She is holding a long, transparent, cylindrical tube horizontally across her chest. The background is dark with blue stage lighting.

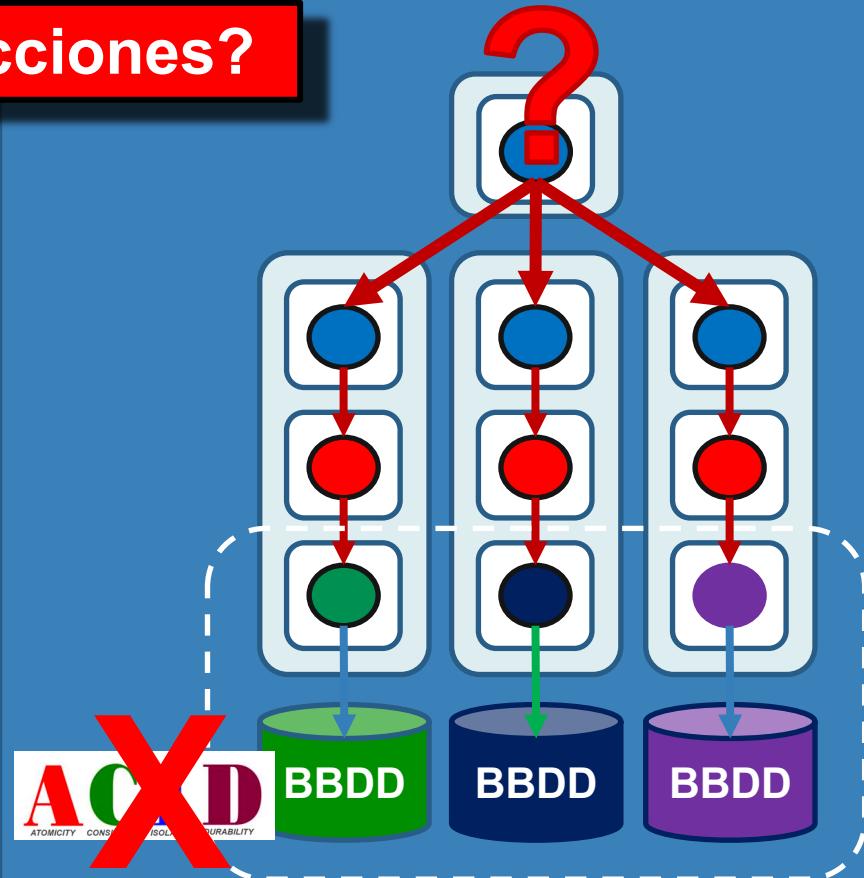
Cuidado con aplicar
DRY entre
Microservicios

Objetivo:
Autonomía

¿Transacciones?



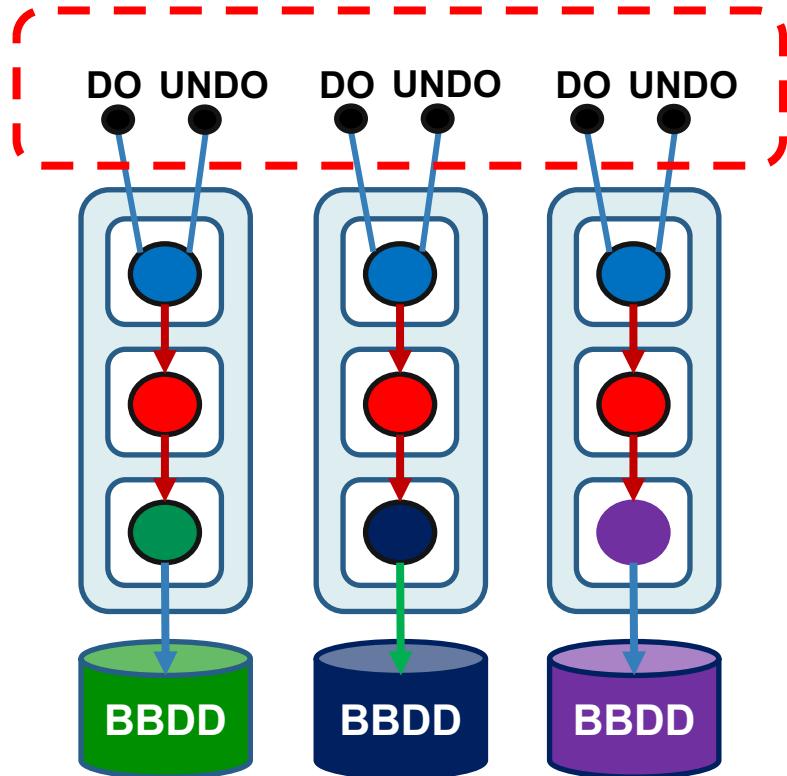
TRADICIONAL



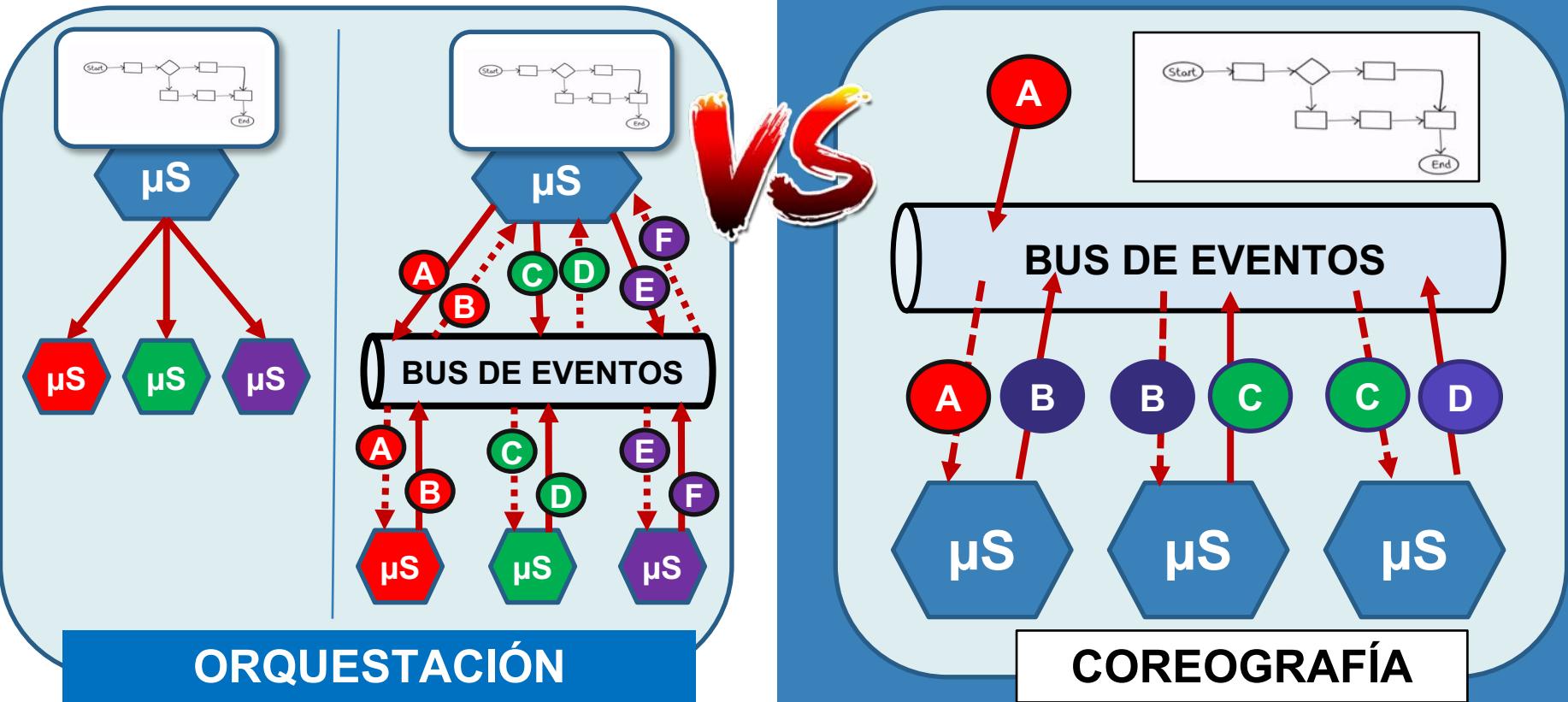
DISTRIBUIDO

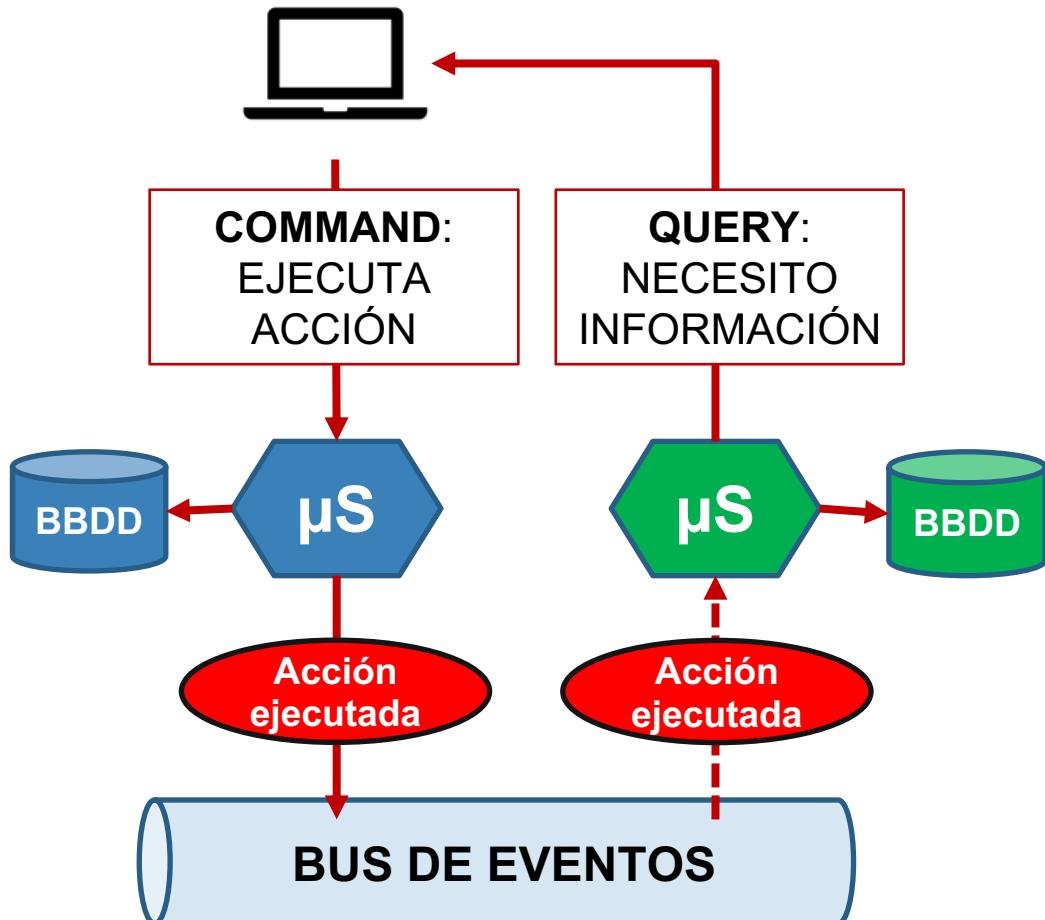


DISEÑO PARA EL FALLO: ACCIÓN (DO) & COMPENSACIÓN (UNDO)



SAGAS





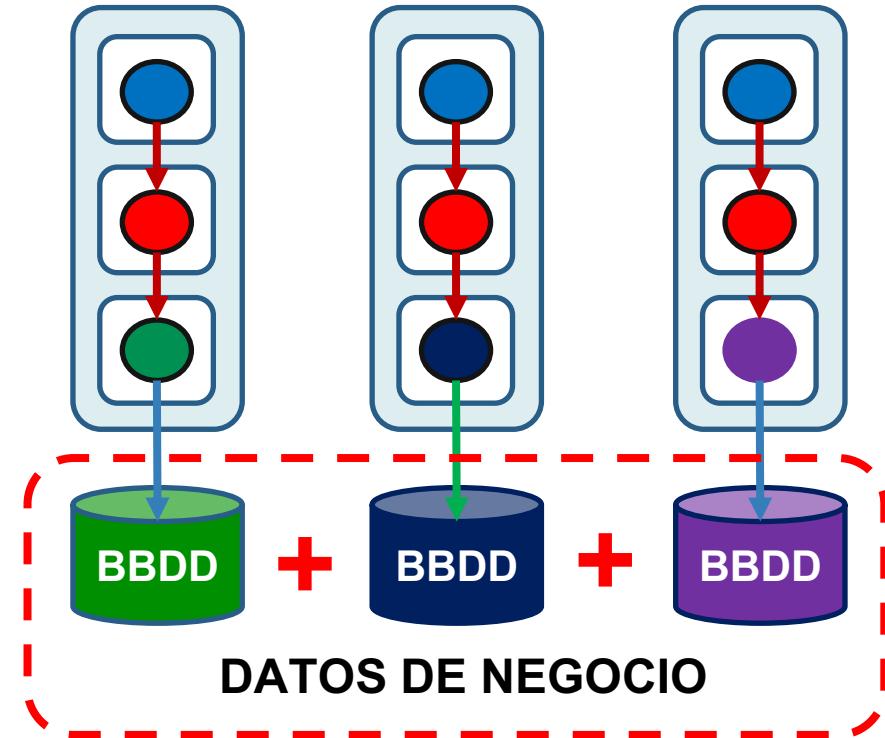
CQRS

LECTURAS VS ESCRITURAS



CONSISTENCIA
EVENTUAL

AGREGACIONES DE DATOS



¿ Join ?



AGREGADOR
DATOS

μ S



“BAJO DEMANDA”

VISTA
MATERIALIZADA

μ S

A

B

C

BUS DE EVENTOS

A

B

C

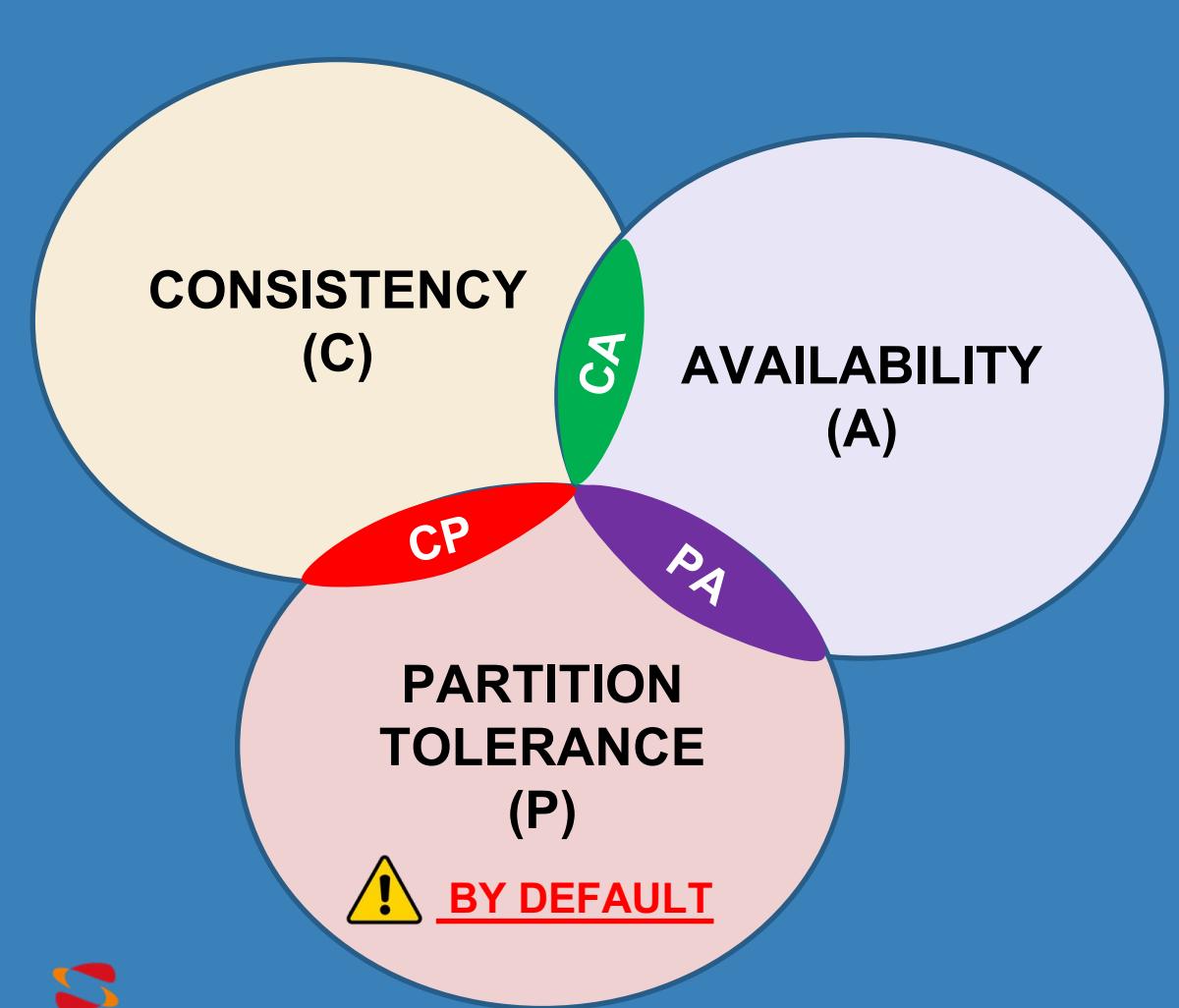
μ S

μ S

μ S



CONSISTENCIA
EVENTUAL



CONSISTENCY
(C)

AVAILABILITY
(A)

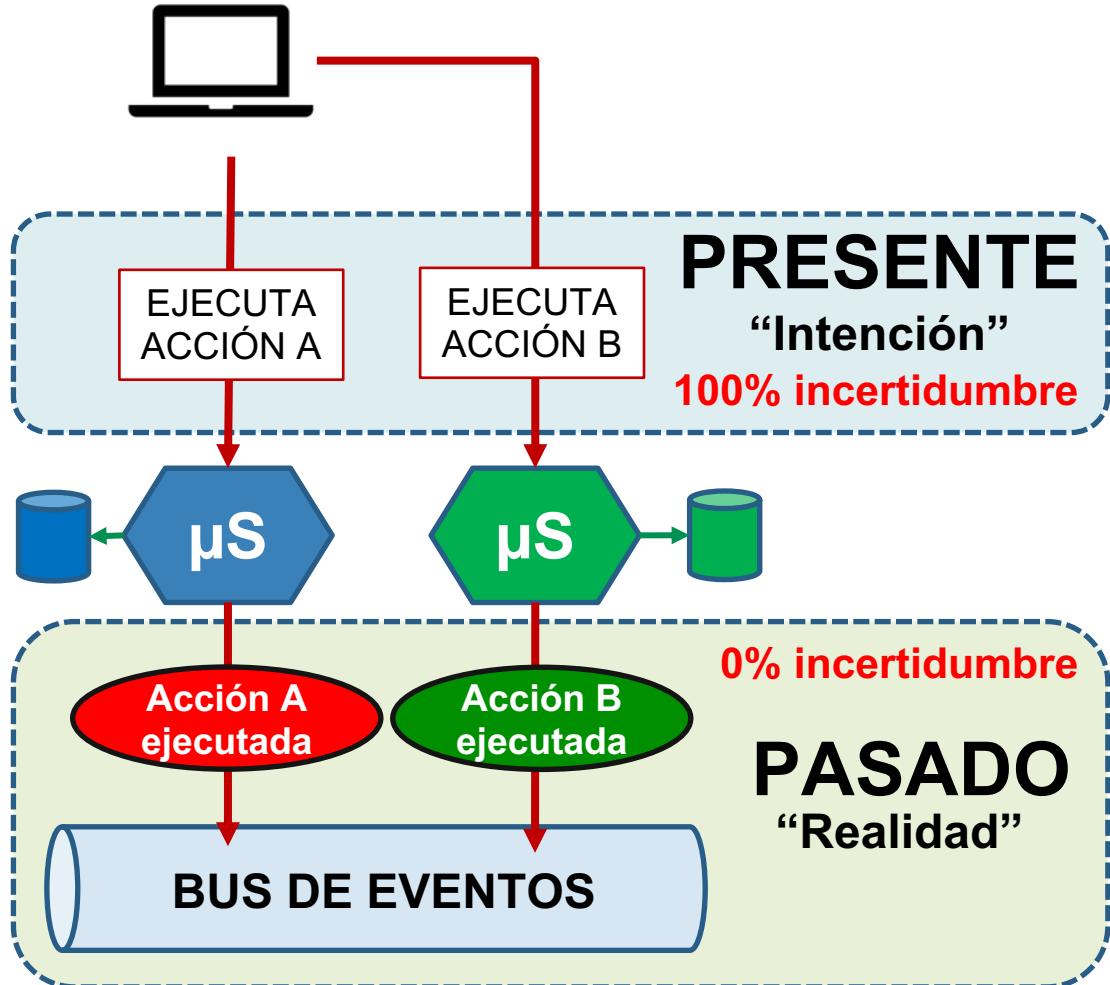
**PARTITION
TOLERANCE**
(P)

⚠ BY DEFAULT

Piensa en las
necesidades de
tu sistema.

Invierte en
diseño



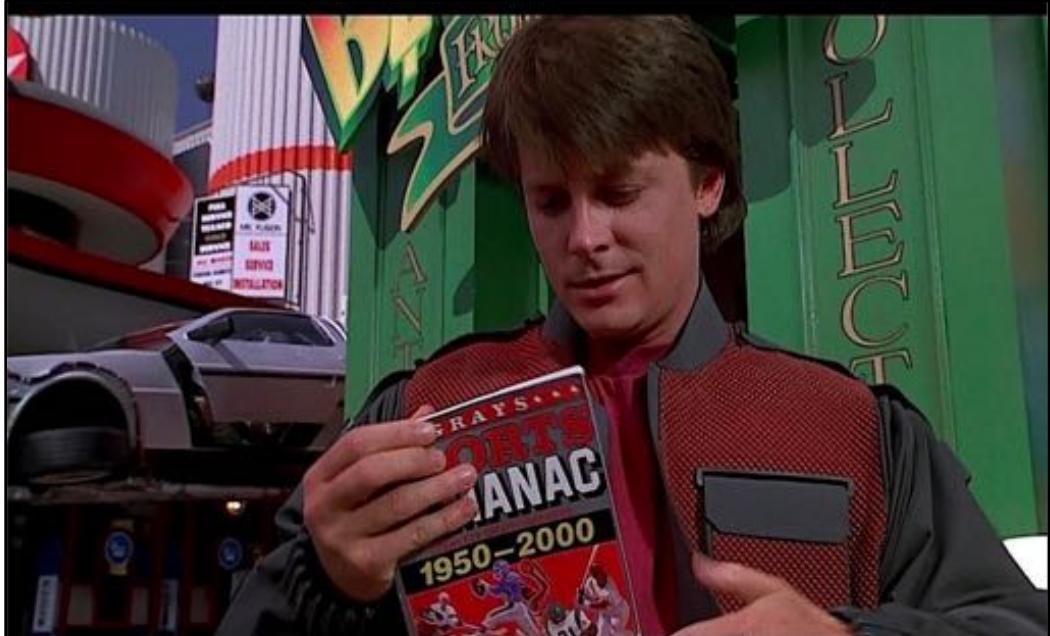


EVENTOS

Los eventos son
tu **REALIDAD**

EVENT SOURCING

¿Imagina que tuvieras un “almanaque” en el que estuvieran registrados todos los eventos de tu sistema?



EVENT SOURCING

Habitualmente trabajas con este concepto aunque no te des cuenta...



Commits on Nov 17, 2018

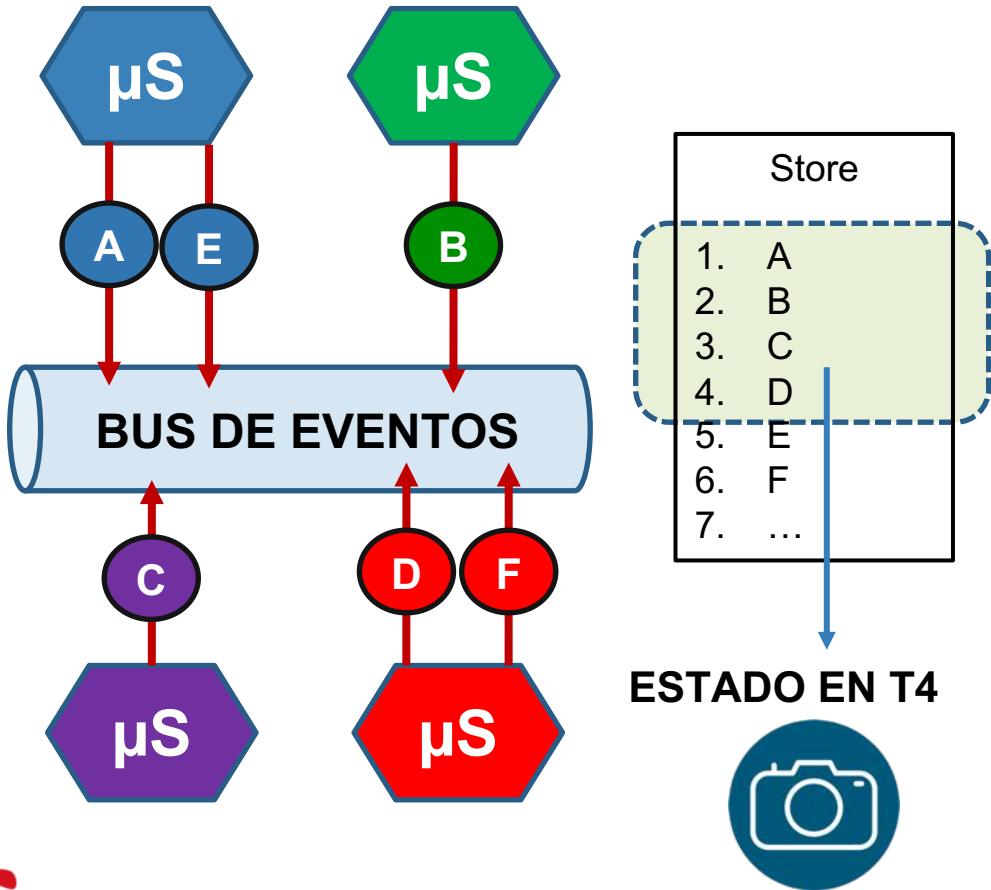
- KAFKA-7402: implement KIP-376 AutoCloseable additions
shunge authored and cmccabe committed 2 days ago
- * MINOR: Catching null pointer exception for empty leader URL when as...
asdf2014 authored and guozhangwang committed 2 days ago

Commits on Nov 16, 2018

- KAFKA-6774: Improve the default group id behavior in KafkaConsumer. (K...
vahidhashemian authored and hachikuji committed 3 days ago)
- KAFKA-7576: Fix shutdown of replica fetcher threads (#5875)
rajinislvaram authored and hachikuji committed 3 days ago

Commits on Nov 15, 2018

- HOTFIX: remove deprecated calls
guozhangwang committed 3 days ago
- MINOR: Remove redundant SuppressIntegrationTests (#5896)
vcephel authored and guozhangwang committed 3 days ago
- MINOR: improve Punctuation JavaDocs and add runtime argument check (#...
mjsax authored and guozhangwang committed 3 days ago)
- KAFKA-7584: StreamsConfig throws ClassCastException if max.in.flight....
mjsax authored and guozhangwang committed 3 days ago



EVENT STORE

El log te da información sobre el estado del sistema y de cómo se ha llegado allí

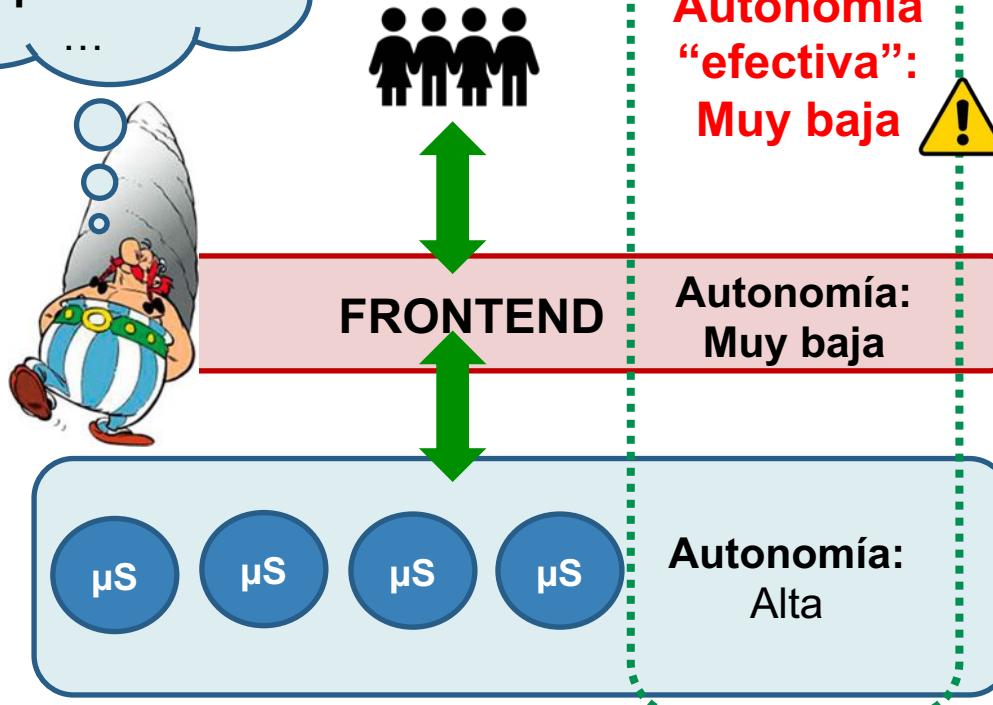


EVENT SOURCING

¿Y si pudieras
repetir el pasado o
crear un **FORK** del
pasado?



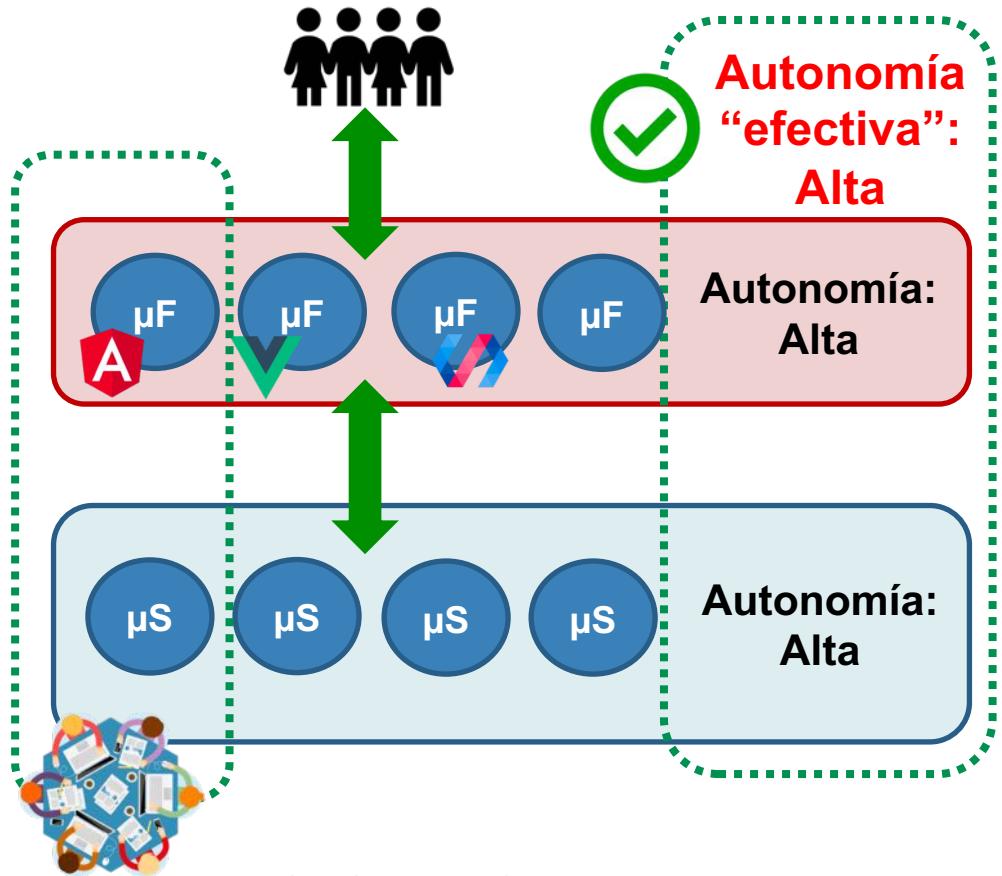
¿Poliglotismo?
¿Despliegue
independiente?
...



FRONTEND

¿Eres
totalmente
autónomo
end-to-end?





AUTONOMIA TOTAL

Equipos y producto end-to-end



FRONTEND

Recuerda:
Autonomía

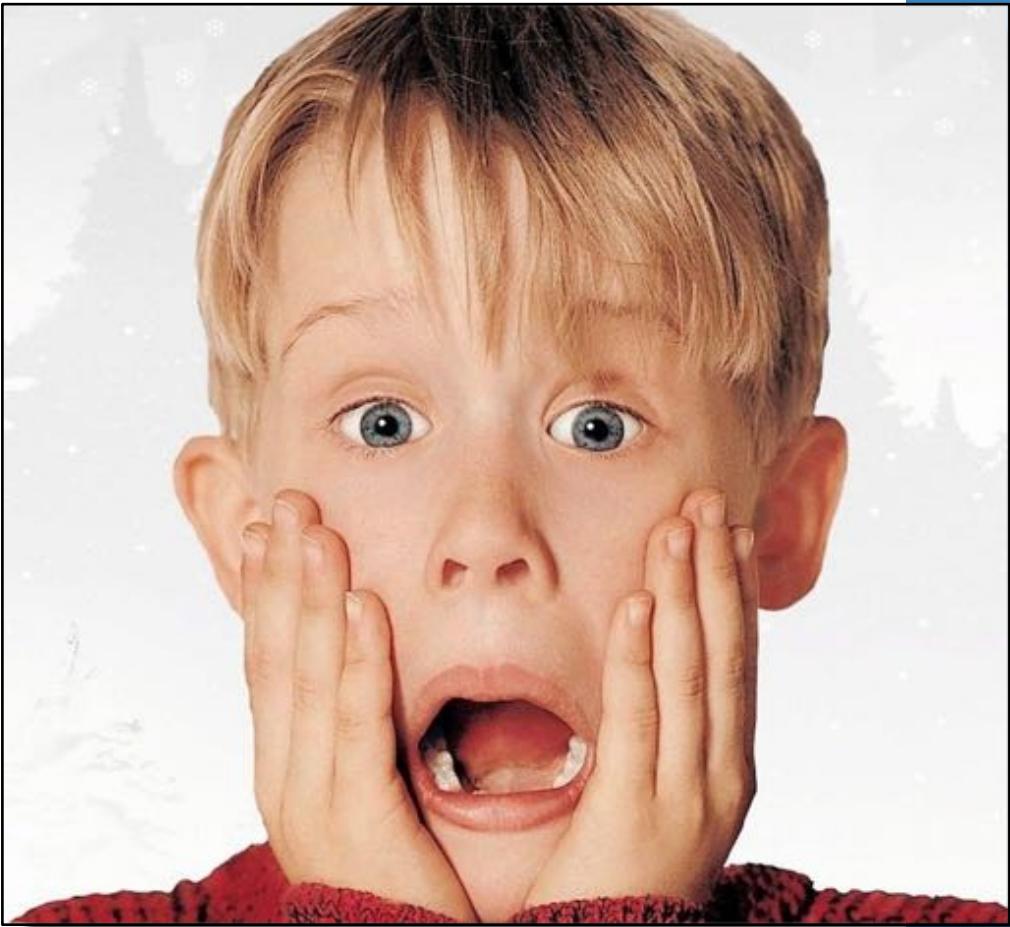
- Dedica **tiempo al diseño y DDD pero sé pragmático**. Nada es blanco o negro.
- **No todo está funcionando mal**, ¿qué necesidades tienes?
- Presta atención a la **idempotencia**.
- **Versionado**: versiona cuando sea necesario, no antes.
- CQRS y las vistas materializadas no son gratis, añaden **complejidad** al sistema. No abuses

ATENCIÓN

“Usar con precaución,
producto no
recomendado para
menores”



¿Sigues
pensando en
Spring Boot?



¿Arquitectos?

¿Quién
necesita
“Arquitectos”?



**¡¡ NO NECESITAS
“GALÁCTICOS” !!**



commit
MAD · NOV 23-24 · 2018

ROGUE ONE

Cuidado con el
“egómetro”.

No estás
buscando héroes





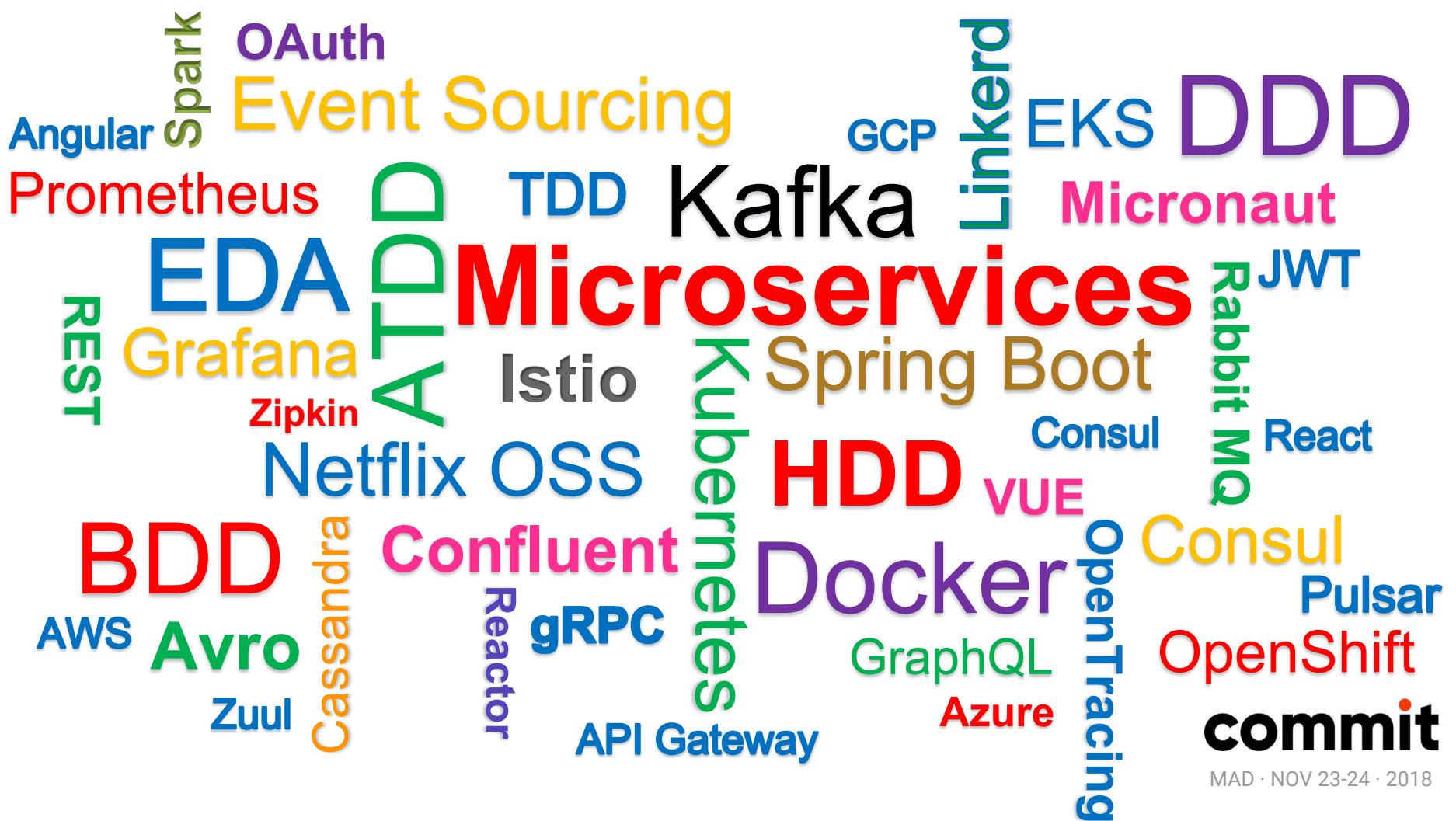
Necesitas
“gente normal”
que baje al
barro

- El objetivo es encontrar la mejor solución al problema completo (end-to-end).
- No hay arquitectura perfecta. Hay arquitectura adaptada al contexto existente en cada momento.
- Personas, equipos y colaboración a todos los niveles
- Arquitectura ágil
- Convivir con el cambio y el fallo

Resumiendo...,

¿qué entendemos
nosotros por
Arquitectura?





Looking for love in all the wrong frameworks



Hype Driven Development

Life on the Bandwagon

O RLY?

@ThePracticalDev

Microservicios

50%
Dto.

Y este **Black Friday**
120 instancias

gratis 1 año

Recuerda:
**ARQUITECTURA alineada
con NEGOCIO y no al revés**

MUCHAS GRACIAS



José Manuel López Doña
CTO (Spain) at Sopra Steria

[@jmlopezdona](https://www.linkedin.com/in/jmlopezdona)



José Alberto Ruiz Casarrubios
Head of Modern Architecture at Sopra Steria

<http://jaruiz.io>
[@jalb80](https://www.linkedin.com/in/jaruizcasarrubios)

sopra steria

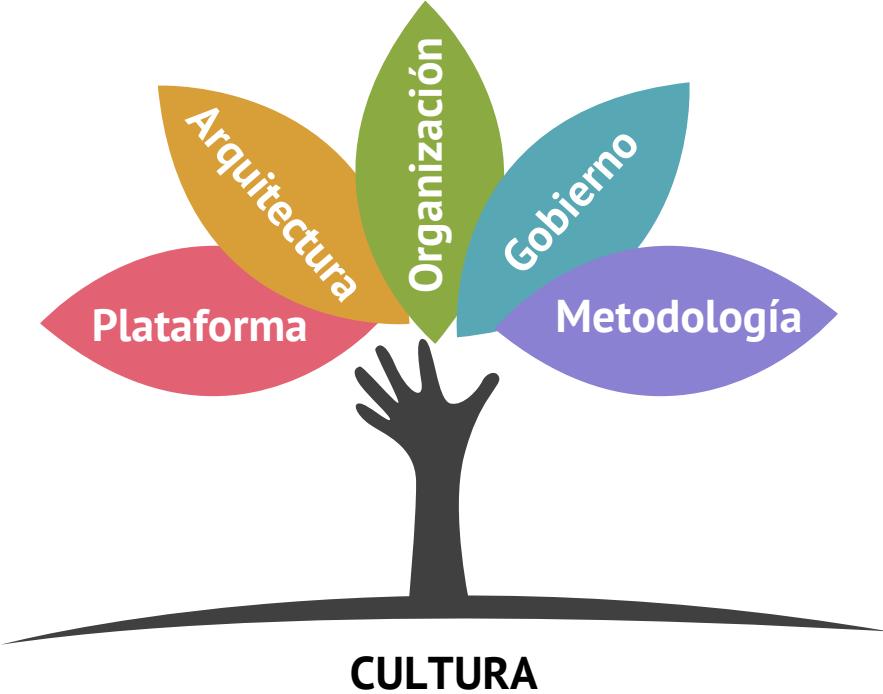
commit
MAD · NOV 23-24 · 2018

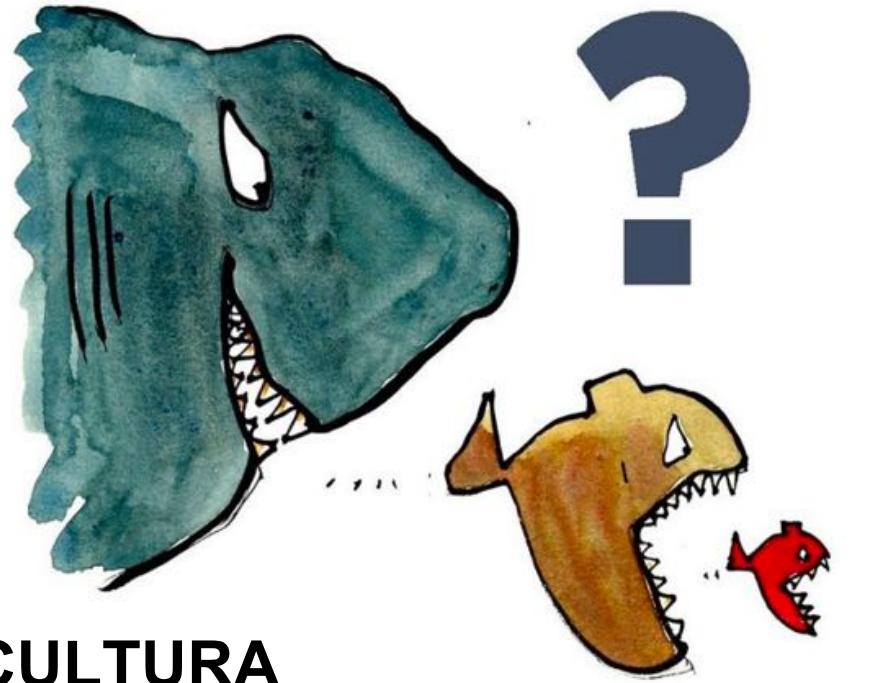


Bonus track

¿qué necesito
para montar?

“Arquitectura”
es sólo una
parte de tu
MONOLITO.





CULTURA

ARQUITECTURA



Puede que
necesites un
CAMBIO
CULTURAL



OLD HABITS

CHANGE

CAMBIA HÁBITOS



- Mejora tu plataforma de desarrollo y ejecución
- Automatiza la construcción y el despliegue tanto de código de aplicación como de infraestructura
- Invierte en pruebas: TDD, BDD, ATDD

No me aplica porque
yo no soy una
compañía de
software...

¿Estás seguro?

CONWAY'S LAW

*"Any organization
that designs a
system will
inevitably produce a
design whose
structure is a copy of
the organization's
communication
structure."*
Melvin E. Conway



commit

MAD · NOV 23-24 · 2018



"Every business will become a software business, build applications, use advanced analytics and provide SaaS services"
(Satya Nadella – CEO Microsoft)



SÉ ÁGIL

El futuro no es
predecible, no lo
conoces, evita
riesgos.



José Alberto Ruiz Casarrubios

Head of Digital Architecture / Hands-on Software Architect

1 semana

Tu pareja, con la que llevas muchos años y conoces bien, te pide pintar la habitación de los niños de azul y rosa.

Te arriesgarías a elegir tono de colores, lo que pintas de cada color y se lo enseñas ya terminado o darías unos brochazos para que vaya viendo el tono y confirme que lo quiere así y con esa distribución?

Imagina ahora que esa habitación es software y encima no conoces el entorno/cliente tan bien como a tu pareja... Esperas al final o lo vas enseñando?

commit

MAD · NOV 23-24 · 2018



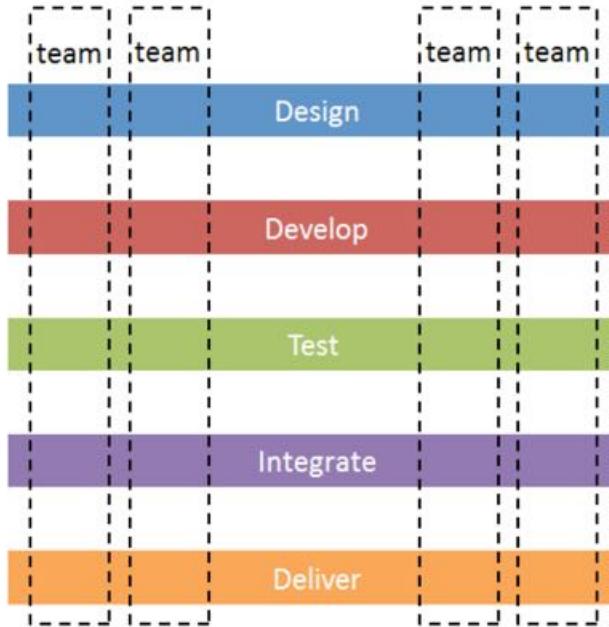
*“The best architectures,
requirements, and
designs
emerge from self-
organizing teams.”*
(Agile Manifesto)



Promueve la autonomía

¿Tus equipos tienen todos
los **skills y capacidades**
necesarias?

Cross Functional



commit

MAD · NOV 23-24 · 2018

MUCHAS GRACIAS



José Manuel López Doña
CTO (Spain) at Sopra Steria

[@jmlopezdona](https://www.linkedin.com/in/jmlopezdona)



José Alberto Ruiz Casarrubios
Head of Modern Architecture at Sopra Steria

<http://jaruiz.io>
[@jalb80](https://www.linkedin.com/in/jaruizcasarrubios)

sopra steria

commit
MAD · NOV 23-24 · 2018