

>_<



Survival Guide for the Java Architect in the Cloud Era

Otávio Santana
@otaviojava
Distinguished Engineer

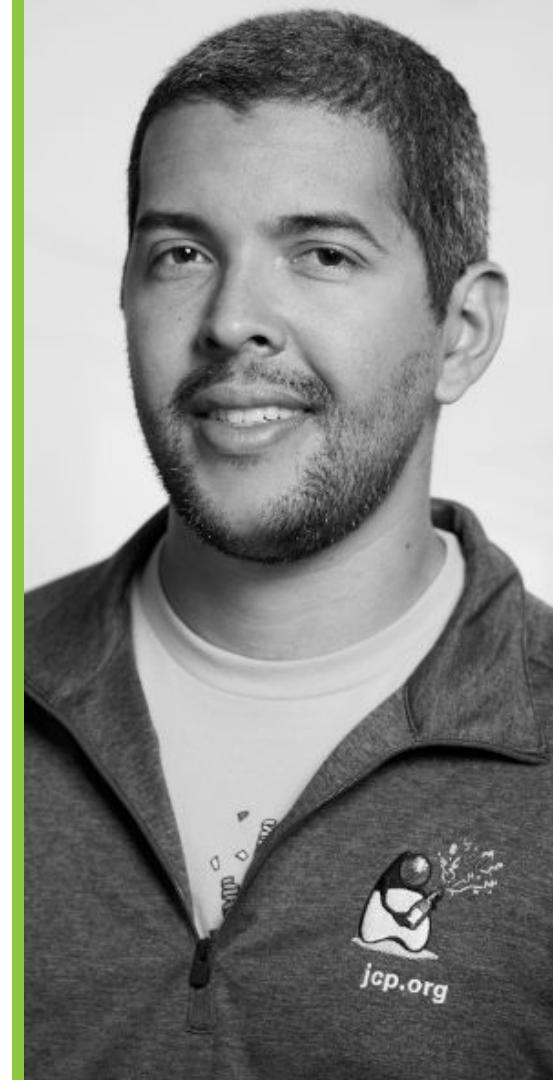
Who am I?

Otavio Santana

Distinguished Engineer

@otaviojava

- Java Champion, Oracle ACE
- JCP-EC-EG-EGL
- Apache and Eclipse Committer
- Jakarta EE and MicroProfile
- CNCF/Linux Foundation/ FINOS Member
- Eclipse Project Leader
- Book and blog writer



<>

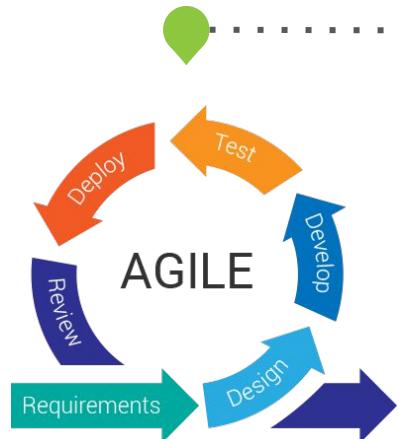


Cloud

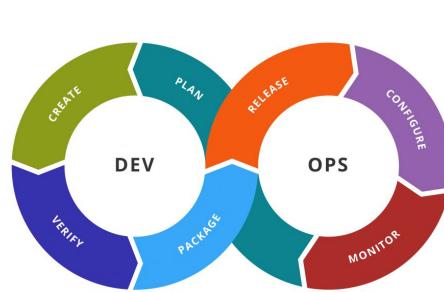


<> History

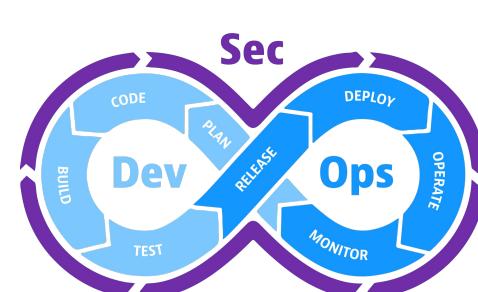
Agile
2001



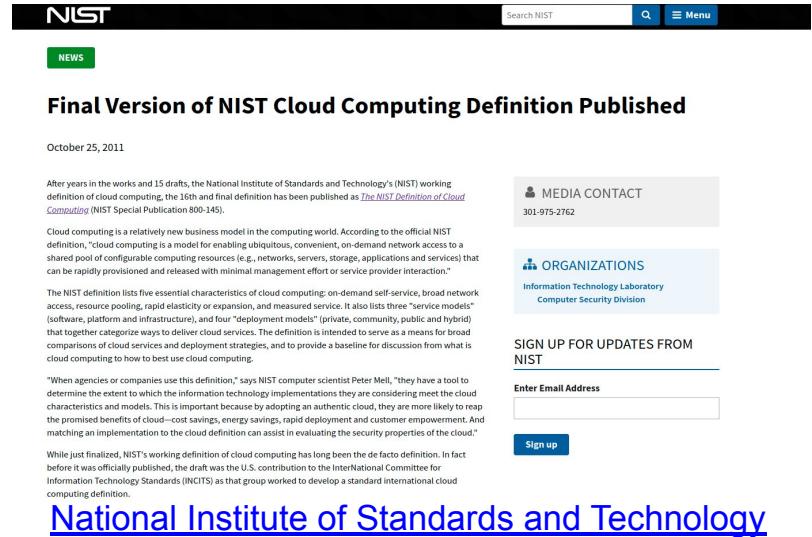
DevOps
2010



DevSecOps
2015



"Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."



The screenshot shows the NIST website's news section. The main headline is "Final Version of NIST Cloud Computing Definition Published" dated October 25, 2011. Below the headline, there is a summary of the definition, a quote from Peter Mell, and a link to the full definition. To the right, there are sections for "MEDIA CONTACT" and "ORGANIZATIONS". At the bottom, there is a sign-up form for updates.

Final Version of NIST Cloud Computing Definition Published

October 25, 2011

After years in the works and 15 drafts, the National Institute of Standards and Technology's (NIST) working definition of cloud computing, the 16th and final definition has been published as *The NIST Definition of Cloud Computing* (NIST Special Publication 800-145).

Cloud computing is a relatively new business model in the computing world. According to the official NIST definition, "cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."

The NIST definition lists five essential characteristics of cloud computing: on-demand self-service, broad network access, resource pooling, rapid elasticity or expansion, and measured service. It also lists three "service models" (software, platform and infrastructure), and four "deployment models" (private, community, public and hybrid) that together categorize ways to deliver cloud services. The definition is intended to serve as a means for broad comparison of cloud services and deployment strategies, and to provide a baseline for discussion from what is cloud computing to how to best use cloud computing.

"When agencies or companies use this definition," says NIST computer scientist Peter Mell, "they have a tool to determine the extent to which the information technology implementations they are considering meet the cloud characteristics and models. This is important because by adopting an authentic cloud, they are more likely to reap the promised benefits of cloud—cost savings, energy savings, rapid deployment and customer empowerment. And matching an implementation to the cloud definition can assist in evaluating the security properties of the cloud."

While just finalized, NIST's working definition of cloud computing has long been the de facto definition. In fact, before it was officially published, the draft was the U.S. contribution to the International Committee for Information Technology Standards (INCITS) as that group worked to develop a standard international cloud computing definition.

[National Institute of Standards and Technology](#)

MEDIA CONTACT
301-975-2762

ORGANIZATIONS
Information Technology Laboratory
Computer Security Division

SIGN UP FOR UPDATES FROM NIST

Enter Email Address

[Sign up](#)

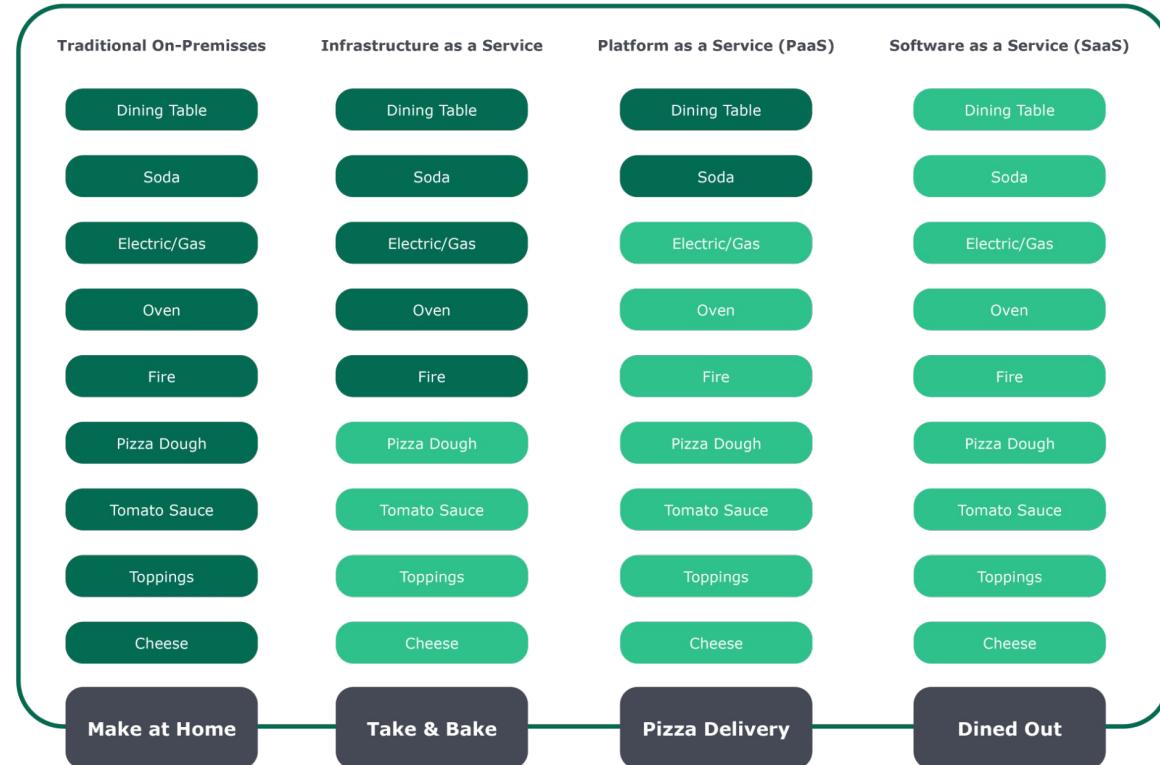
<> Someone else's problem



- Hardware
- Operations
- Knowledge



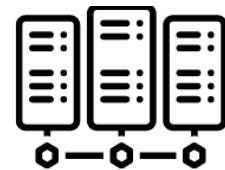
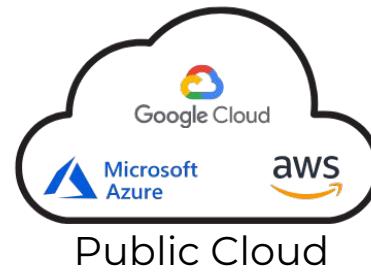
Pizza as a Service



<> Success: Faster innovation, without impacting stability



>_< Cloud flavors

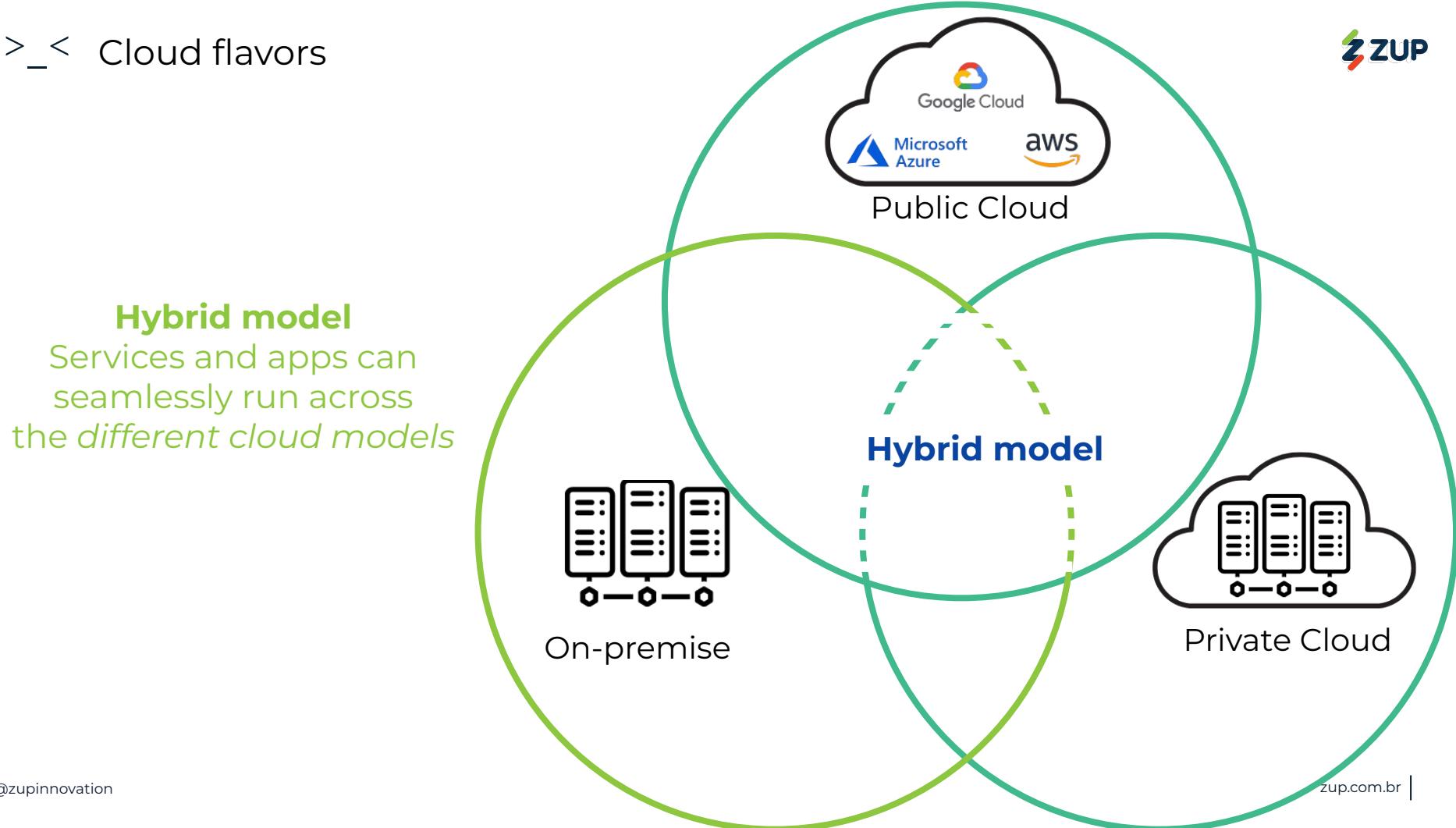


On-premise

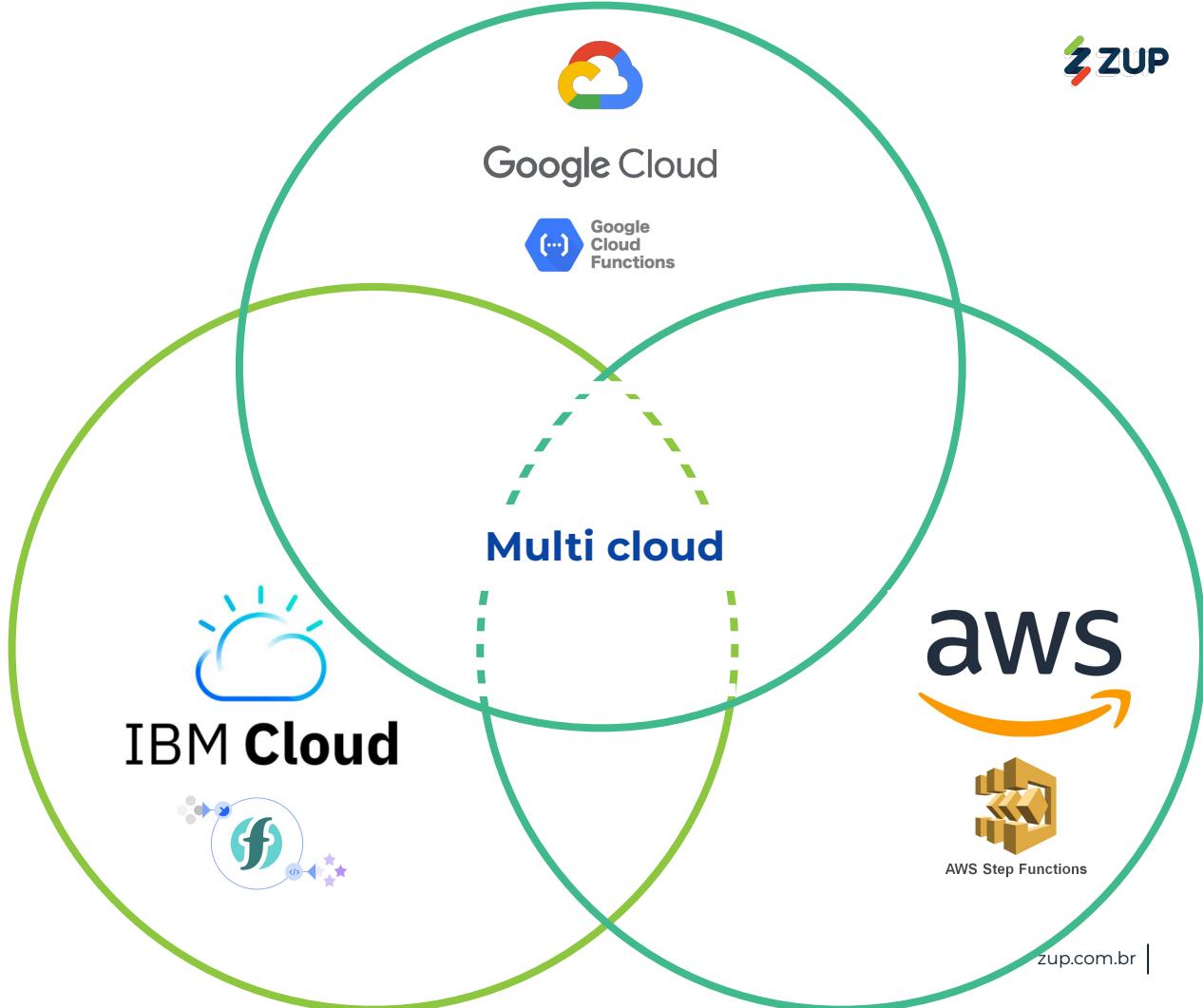


Private Cloud

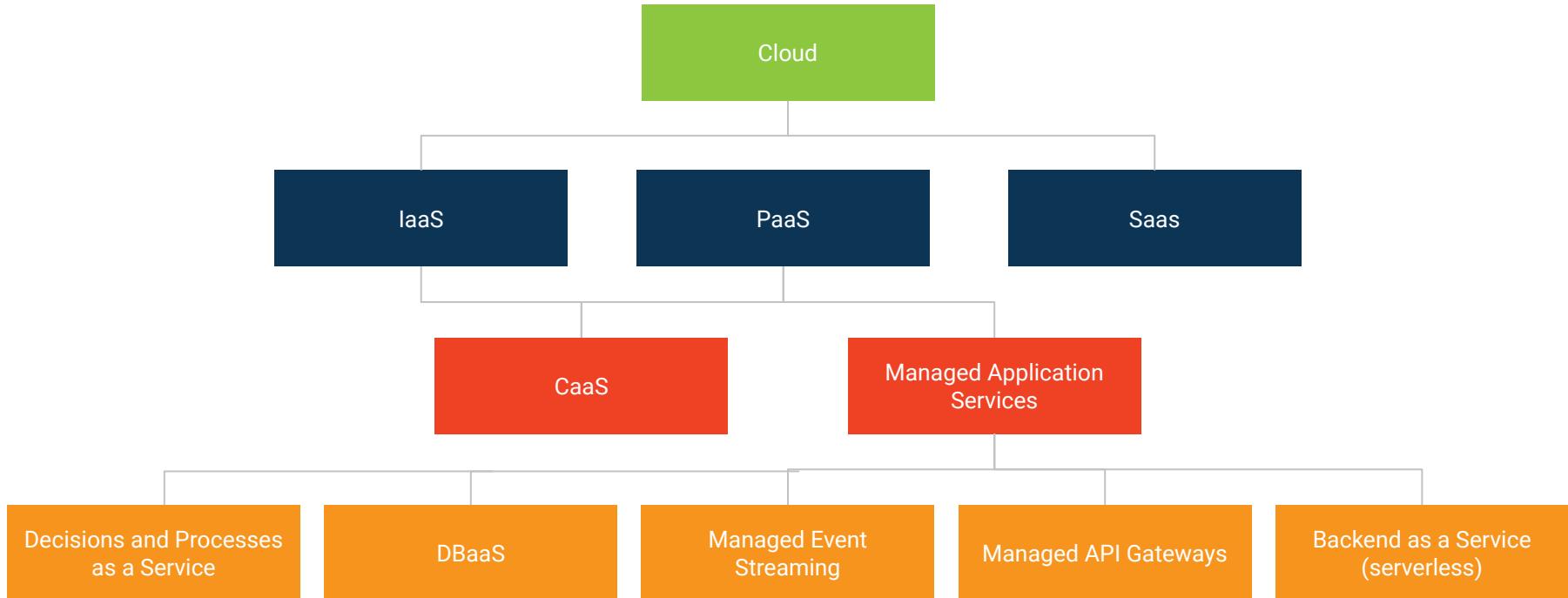
>_< Cloud flavors



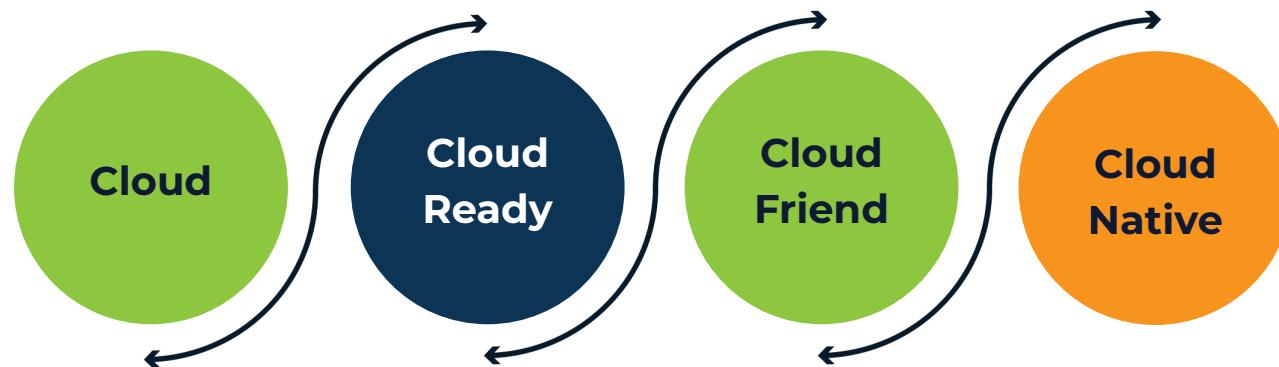
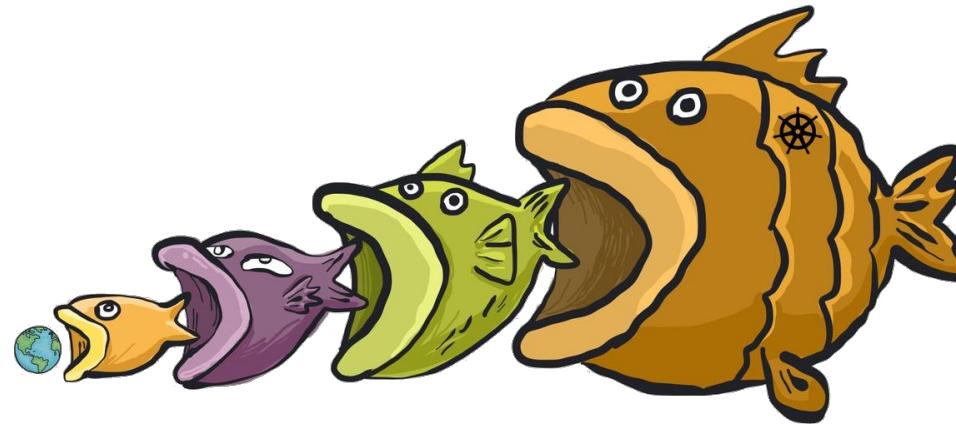
Multi-cloud model
Usage of two or more
vendors of a same
cloud service type



< > More Options...



<> Cloud Trends



Cloud Native Applications

“Cloud-native is an **approach** to
building and **running** applications that
exploits the **advantages of the cloud**
computing model.”



The screenshot shows the Pivotal website, which is now part of VMware. The main navigation bar includes links for Why Pivotal, Kunden, Produkte und Dienstleistungen, Ressourcen, Unternehmen, Support, Anmelden, and a search icon. The page features a dark header with the text "TECH INSIGHTS" and "Cloud-Native Applications: Ship Faster, Reduce Risk, Grow Your Business". Below the header is a graphic of a green and red grid with a play button icon. The main content area has a dark background with text links for "Cloud-Native Apps", "Perspectives", "At Pivotal", "Customer Stories", and "Next steps".

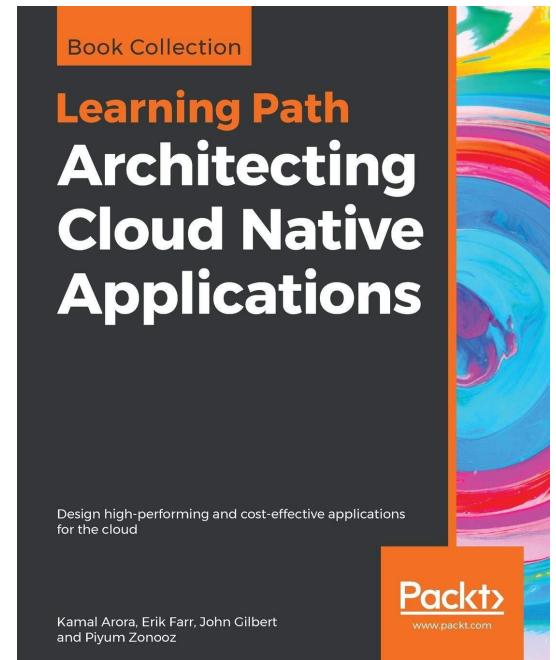
What are Cloud-Native Applications?

Cloud-native is an approach to building and running applications that exploits the advantages of the cloud computing model. The cloud has redefined the competitive landscape across virtually all industries by eliminating the focus on capital investment and staff to run an enterprise data center, replacing it with limitless computing power on-demand and pay-as-you-go. Reduced IT spend means a lower barrier to entry with competitive advantage becoming a function of the speed teams can bring new ideas to market, which is why software is eating the world and startups are using cloud-native approaches to disrupt traditional industries.

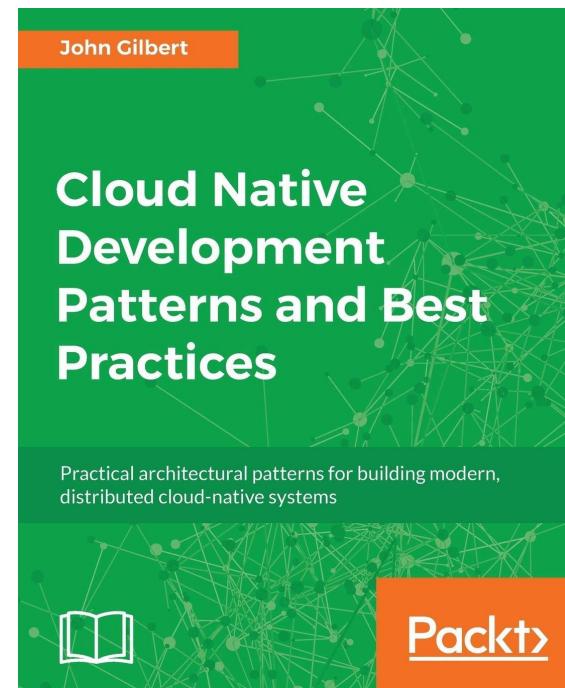
Yet organizations require a platform for building and operating cloud-native applications and services that automates and integrates the concepts of DevOps, continuous delivery, microservices, and containers:



“Cloud-native is **a different way of thinking and reasoning about software systems**. It embodies the following concepts: **powered by disposable infrastructure**, composed of bounded, scales globally, embraces disposable architecture.”



“Independent **DURS** ultimately comes up in every discussion on cloud-native concepts; to **independently Deploy, Update, Replace and Scale.**”



“Cloud-native is **more than a tool set**. It is a **complete architecture**, a **philosophical approach** for building applications that **take full advantage of cloud computing**.”

O'REILLY®

Cloud Native Transformation

Practical Patterns for Innovation

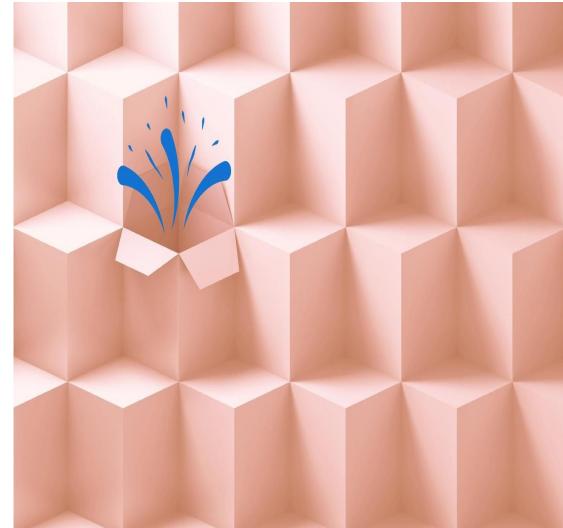


Pini Reznik,
Jamie Dobson &
Michelle Gienow

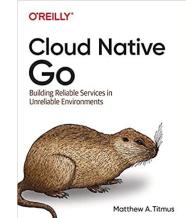
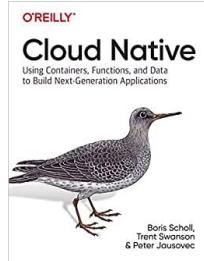
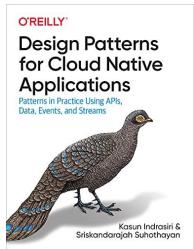
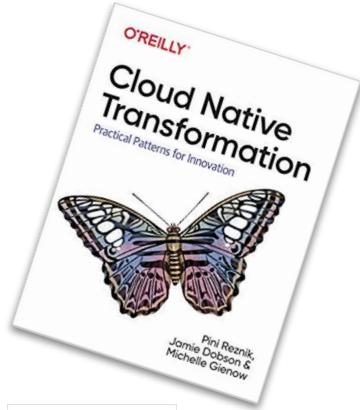
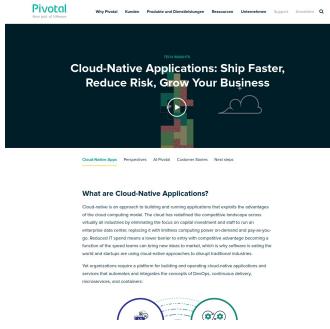
“Cloud-native technologies **empower organizations** to **build and run scalable applications** in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.”



Cloud-native is a term used to describe
container-based environments.



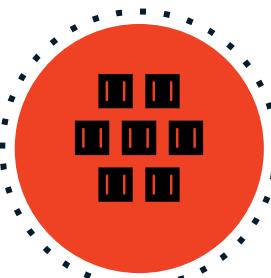
What is the definition for "cloud-native"?



A set of good practices to optimize an application in the cloud through:



Containerization



Orchestration



Automation

< > Cloud: Why do we need it?



Expectation



Reality

Challenges



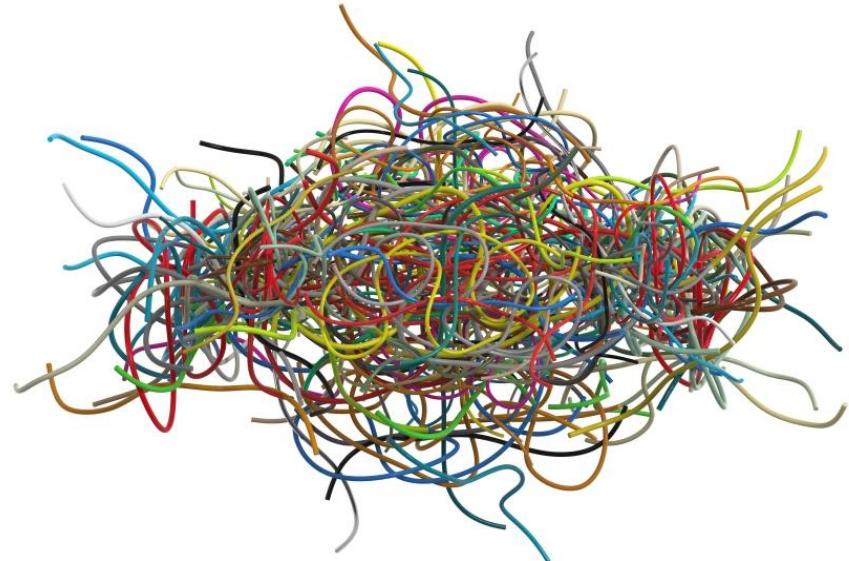
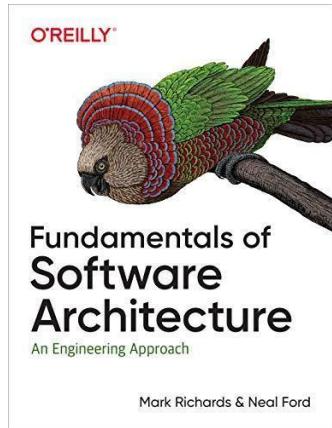
<> Challenges

- . Routes
- . Security
- . Upgrade database and language
- . Backups
- . CI/CD



< > Trade-off

- Complexity
- Risk
- Cost
- Knowledge management



Survival Guide



The Flightplan



What is a flightplan?

- Route (departure, stops, landing, height)
- Schedule (proposed departure and landing time)
- Aircraft type and identification
- Weather
- Equipments
- Flight crew
- Fuel planning
- ETOPS (Extended-range Twin-engine Operational Performance Standards)



What is a flightplan?

- where am I?
- How far I want go?
- Internal conditions
- External conditions
- Budget
- Safety



Ask yourself:

"Where are we?"

"Where should we go next?"

"How do I get there?"



DESCRIPTION

● Information technology

● Project Management

● Business Processes

● Diverse

Cloud
Maturity
Model

Richardson
Maturity
Model

CMM

Social media
Innovation

Maturity Model for
Knowledge
Management
Strategic Benefits

Kubernetes
Maturity
Model

Consumer
Cloud
Maturity

Scaled Agile
Framework
(SAFe)

Portfolio,
Programme and
Project Mgmt
Maturity Model
(P3M3)

Process
Management in
Hospitals

Business
Process
Management
Maturity Model

Glory of REST



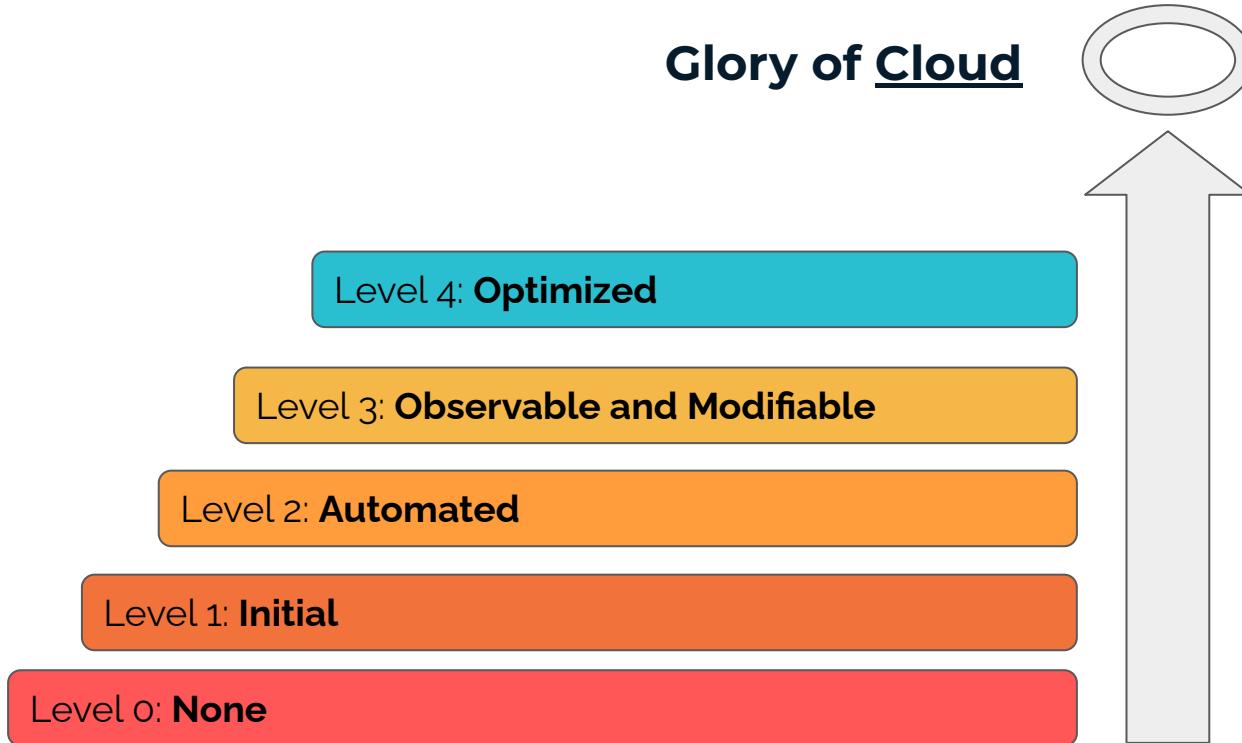
Level 3: Hypermedia Controls

Level 2: HTTP Verbs

Level 1: Resources

Level 0: The Swamp of POX

Glory of Cloud



Flightplan

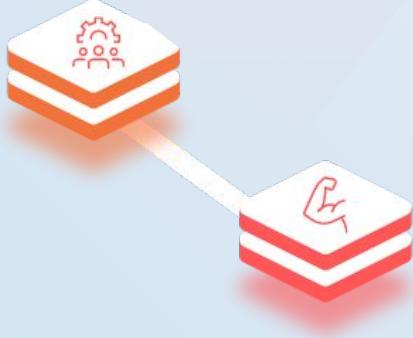


Level 0: None

There's no cloud adoption nor the usage of any practices in development and deployment cycles.



Flightplan



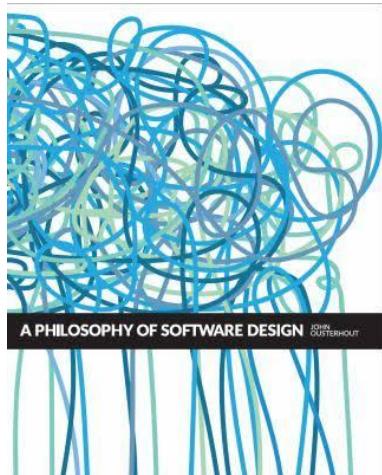
Level 01: Initial

There's some level of cloud adoption, although, automation and observability are low or nonexistent.

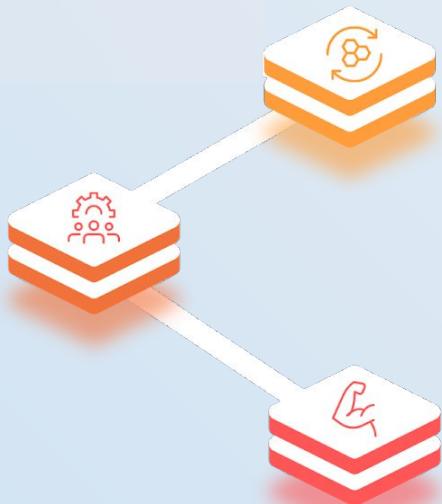
Level 00: None

There's no cloud adoption nor the usage of any practices in development and deployment cycles.

- Risk vs Cost
- Complexity



Flightplan



Level 02: Automated

Deployment of software and infrastructure becomes automated and therefore repeatable. Usage of CI/CD practices becomes natural on cloud environments.

Level 01: Initial

There's some level of cloud adoption, although, automation and observability are low or nonexistent.

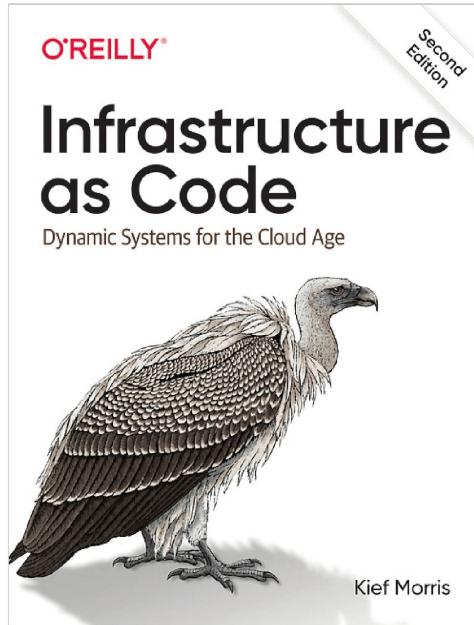
Level 00: None

There's no cloud adoption nor the usage of any practices in development and deployment cycles.

Transparency

Observable

Testable



Security

Modifiable

Reversible

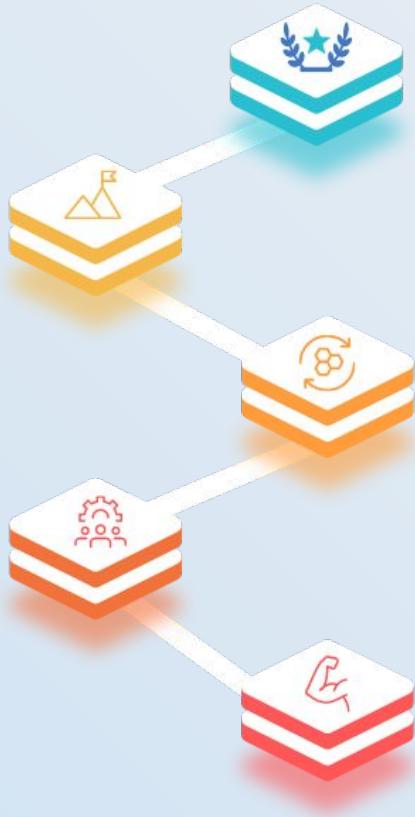
Git as the **single source** of truth

Git as the single place we **operate**



All changes are **observable/verifiable**

Flightplan



Level 04: Optimized

Hybrid cloud is used without affecting monitoring and observability - technical and business KPIs exists. All previous levels applies.

Level 03: Observable and Modifiable

New apps born targeting the cloud. Existing environment can be easily monitored and troubleshooted. Automation is naturally part of the app lifecycle.

Level 02: Automated

Deployment of software and infrastructure becomes automated and therefore repeatable. Usage of CI/CD practices becomes natural on cloud environments.

Level 01: Initial

There's some level of cloud adoption, although, automation and observability are low or nonexistent.

Level 00: None

There's no cloud adoption not the usage of any practices in development and deployment cycles.

< > Tips

**Use Managed Services,
Please (Dan Moore)**

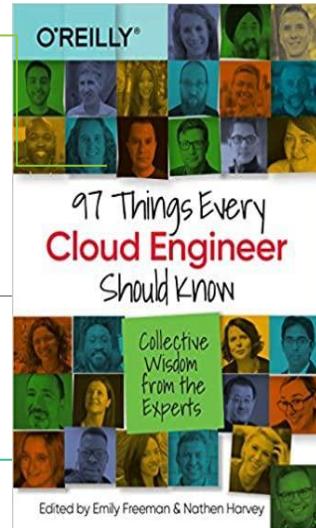
KISS It (Chris Proto)

Source Code Management for
Software Delivery
(Tiffany Jachja)

**It's Ok if you're not
running Kubernetes
(Mattias Geniar)**

Why every Engineer Should Be
a cloud Engineer
(Michelle Brenner)

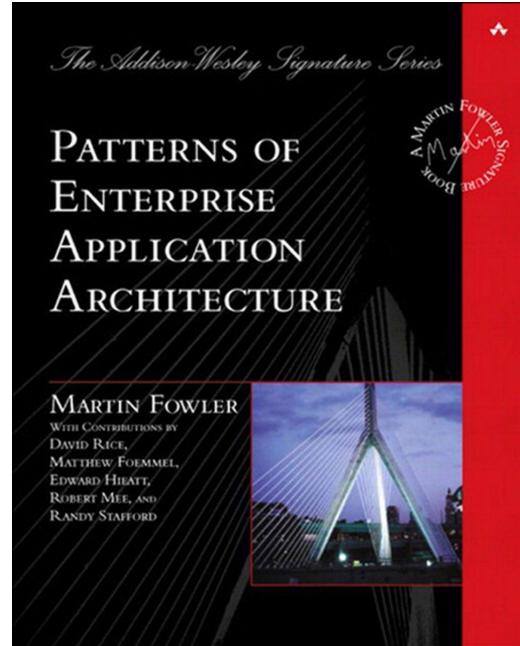
Containers aren't Magic
(Katie McLaughlin)



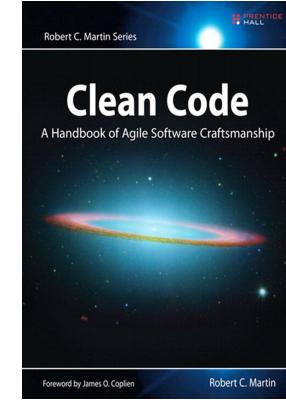
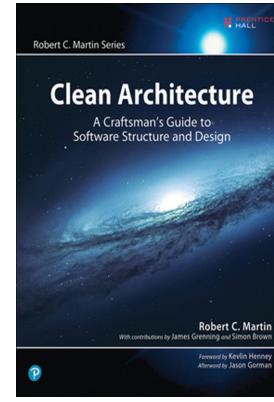
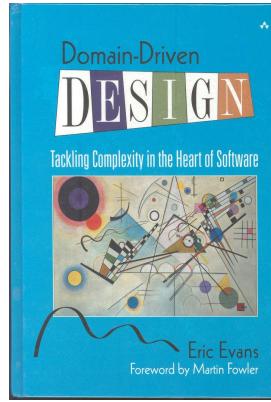
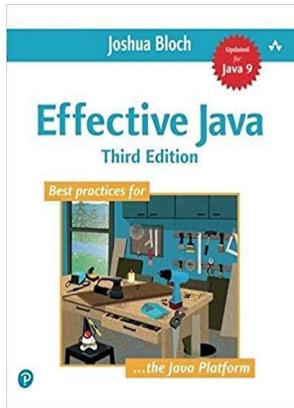
<> The 12 Factors App



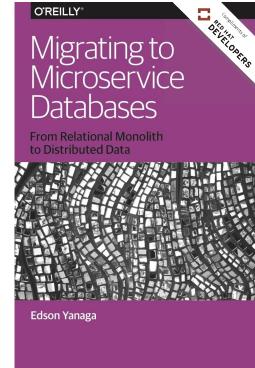
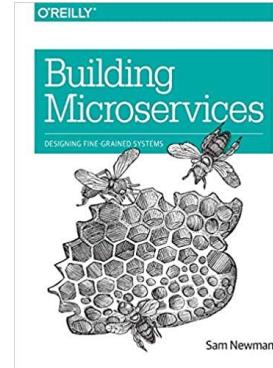
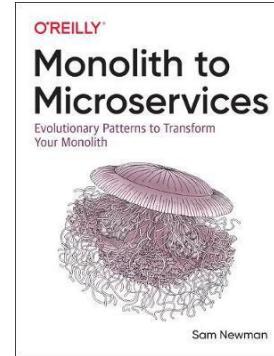
1. Codebase
2. Dependencies
3. **Config**
4. Backing services
5. Build, release, run
6. Process
7. Port binding
8. Concurrency
9. Disposability
10. Dev/prod parity
11. Logs
12. Admin processes



<> Old, but gold



“Certainly, we always read great things about the microservices architectures implemented by companies like Netflix or Amazon. So let me ask a question: how many companies in the world can be Netflix and Amazon?”



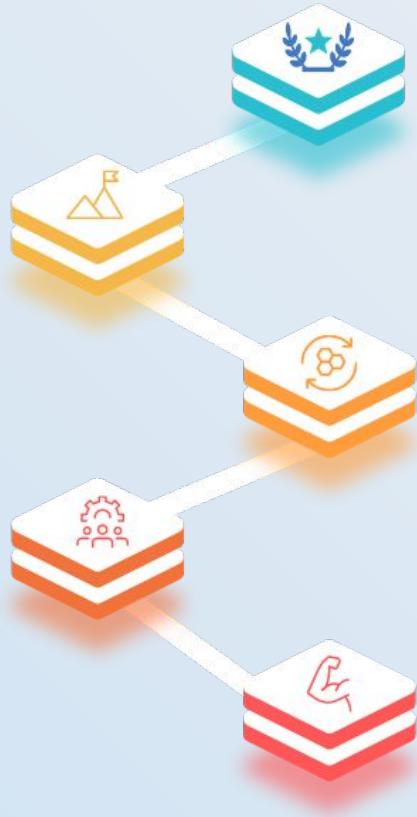
Conway's law

<>

Best Practices in Software Architecture in the Cloud era



Flightplan



Level 04: Optimized

Hybrid cloud is used without affecting monitoring and observability - technical and business KPIs exists. All previous levels applies.

Level 03: Observable and Modifiable

New apps born targeting the cloud. Existing environment can be easily monitored and troubleshooted. Automation is naturally part of the app lifecycle.

Level 02: Automated

Deployment of software and infrastructure becomes automated and therefore repeatable. Usage of CI/CD practices becomes natural on cloud environments.

Level 01: Initial

There's some level of cloud adoption, although, automation and observability are low or nonexistent.

Level 00: None

There's no cloud adoption not the usage of any practices in development and deployment cycles.

Thank you!

Otávio Santana
@otaviojava
Distinguished Engineer

@ZupInnovation