CLOUD NATIVE APPLICATIONS
CONTAINERS, MICROSERVICES, PLATFORMS,CI-CD… OH MY!!

FABIO CHIODINI
Principal System Engineer
@FabioChiodini

WHY ARE WE HERE?

SOFTWARE IS TRANSFORMING INDUSTRIES

NETFLIX	ENTERTAINMENT $53BN	Square	FINANCIAL SERVICES $6BN
airbnb	HOTEL $26BN	TESLA	AUTOMOTIVE $34BN
UBER	TRANSPORTATION $50BN	nest	INDUSTRIAL PRODUCTS $3.2BN

ENTEPRISES ARE FOLLOWING

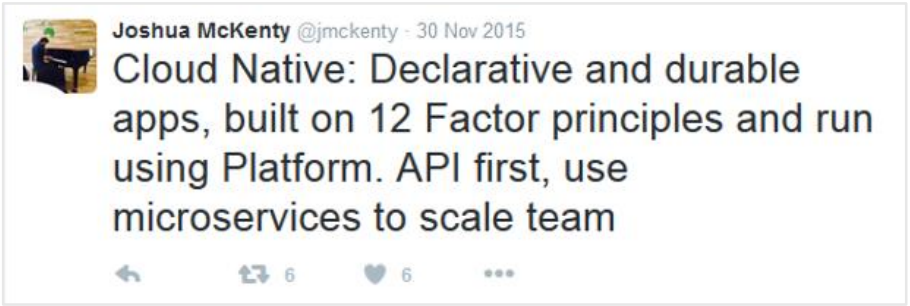
Comcast	700+ apps
Lockheed Martin	Spring Framework + Pivotal CF as a Cloud Native Platform
dish Network	Re-writing software the modern way
The Home Depot	Agile software transformation
Kroger	DevOps adoption with Pivotal CF automated build pipeline
Allstate	Major IT transformation
Mercedes-Benz	Connected cars and Smart Apps
BOSCH	Bosch IoT Suite

CLOUD NATIVE APPLICATIONS

"Applications that do not require resilient infrastructure"

Cloud native is a term describing software designed to run and scale reliably and predictably on top of potentially unreliable cloud-based infrastructure.

Cloud-native applications are purposefully designed to be infra-structure unaware, meaning they are decoupled from infrastructure and free to move as required.*



Cloud Native Apps: The modern, modular approach to to building applications that are focused on web-scale, mobile first and real-time data.

*Duncan C.E. Winn

HOW DO THE TERMS FIT IN YOUR LIFE-CYCLE?

DevOps

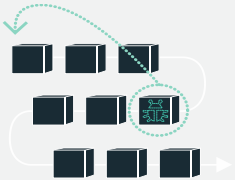
NOT MY PROBLEM
Separate tools, varied incentives, opaque process

SHARED RESPONSIBILITY
Common incentives, tools, process, and culture

CI/CD

RELEASE ONCE EVERY 6 MONTHS
More bugs in production

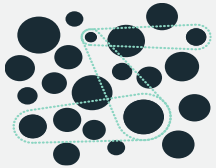
RELEASE EARLY AND OFTEN
Higher quality of code



Microservices

TIGHTLY COUPLED COMPONENTS
Slow deployment cycles waiting on integrated test teams

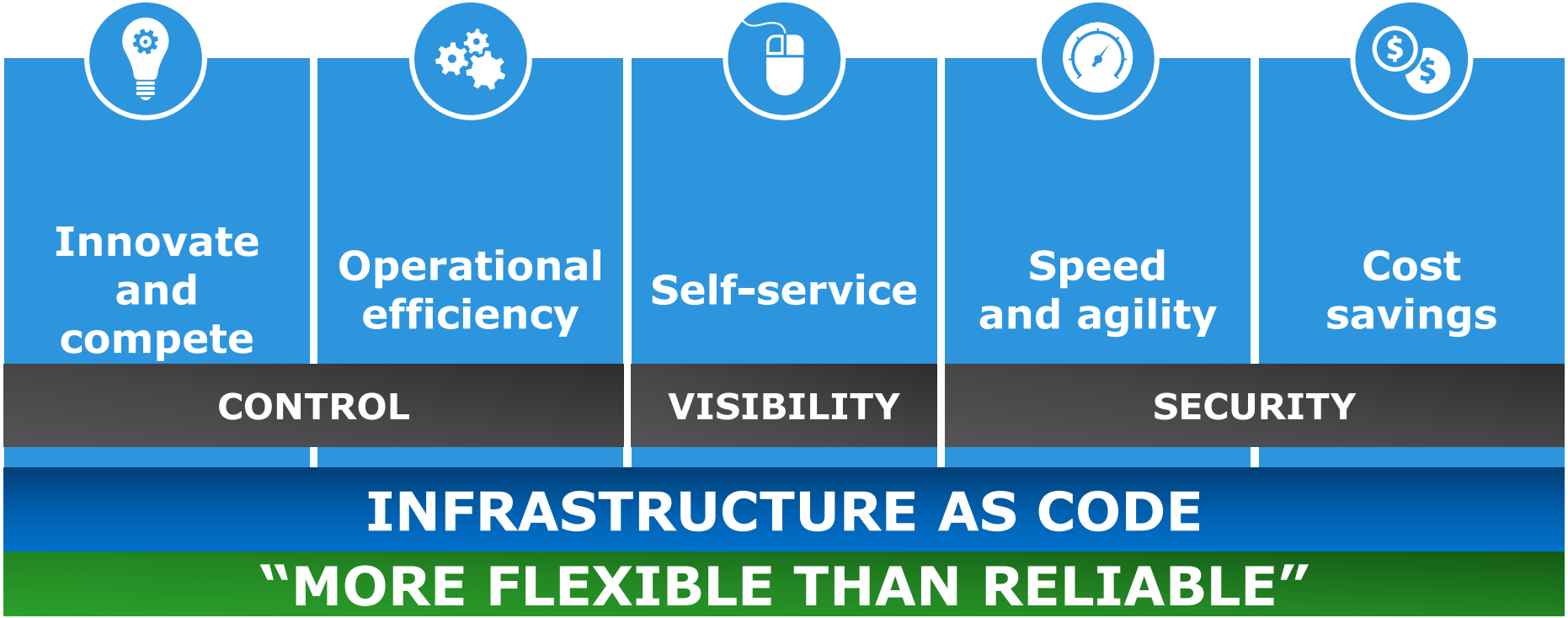
LOOSELY COUPLED COMPONENTS
Automated deploy without waiting on individual components



MANAGE **DEPLOY** **DESIGN**

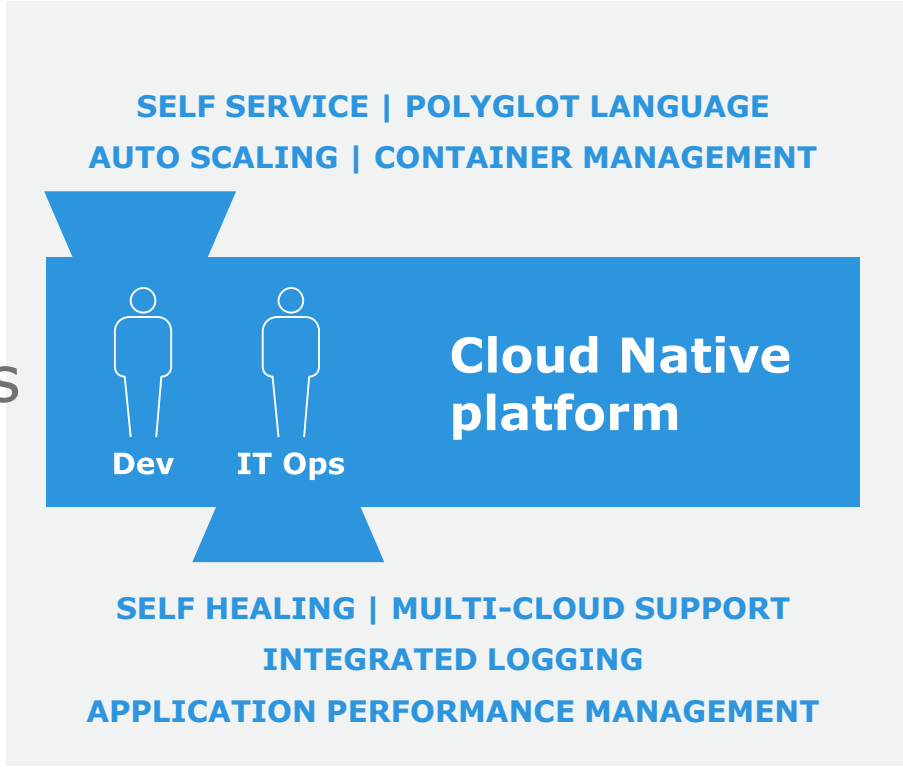
RESULTING IN NEW REQUIREMENTS FOR IT

TO DEPLOY AND DELIVER APPLICATIONS RELIABLY, AT SCALE



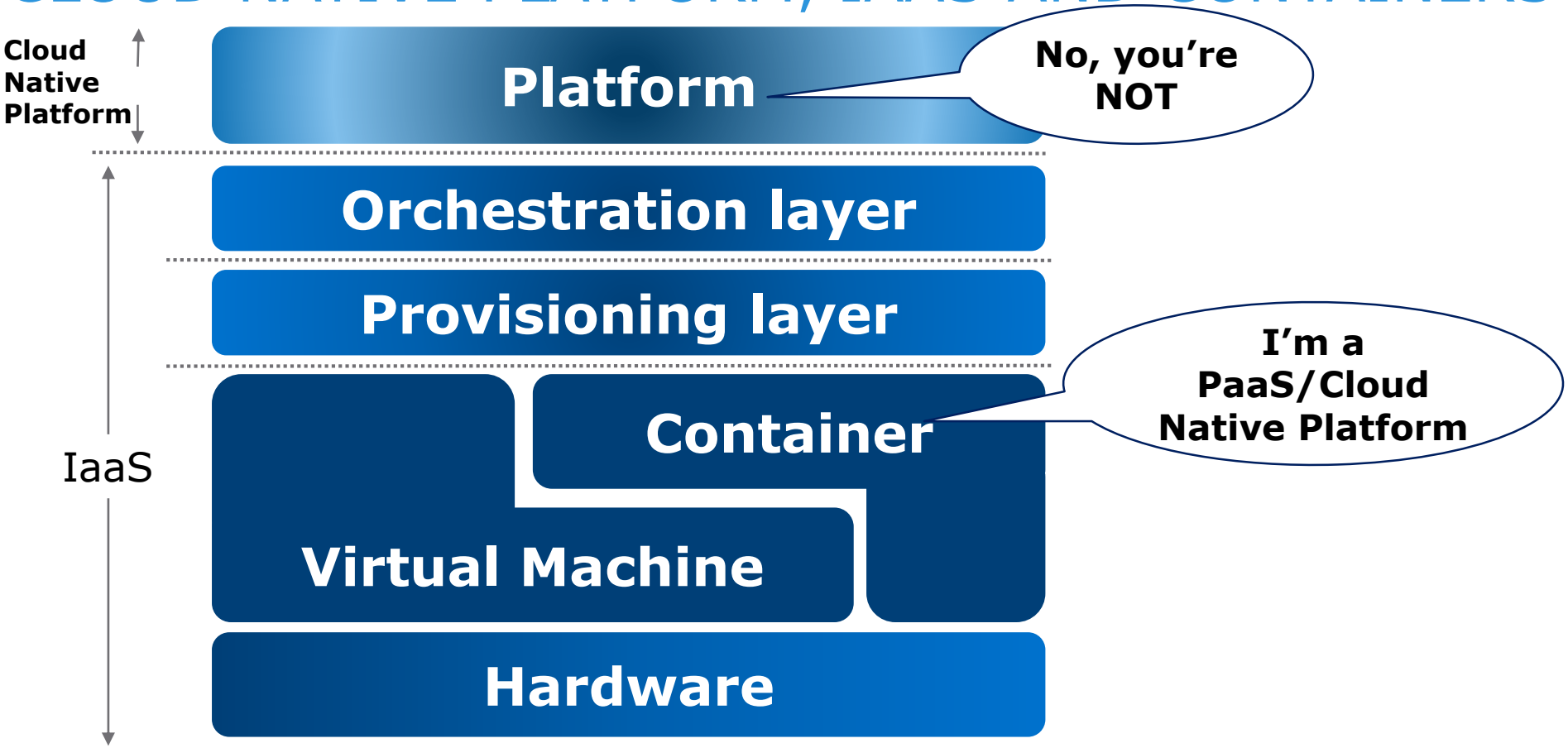
CLOUD NATIVE PLATFORM REQUIREMENTS

- **Programmability** ("Infrastructure As Code")
- **Elasticity** (Which Demands A Scale-Out Architecture)
- **Economics** (Steers Towards Standard Servers + Software)
- **Strong Instrumentation And Telemetry** Of Infrastructure Layer



BTW…
CONTAINERS vs VMs 101

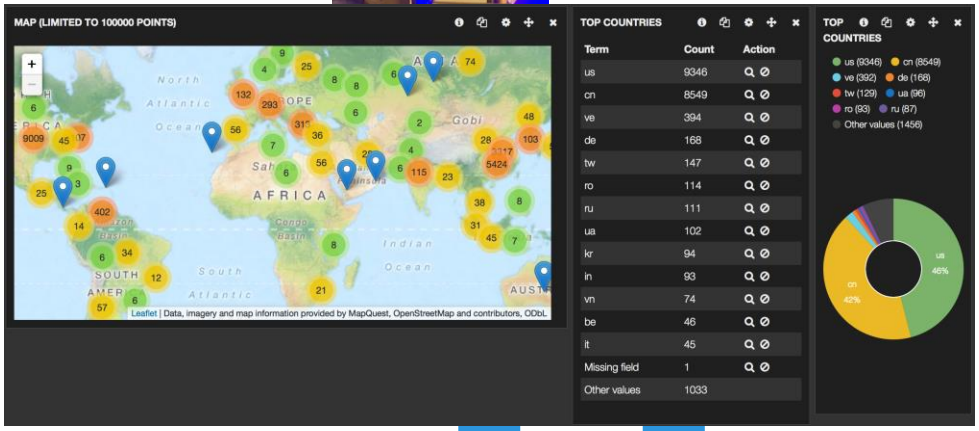
CLOUD NATIVE PLATFORM, IAAS AND CONTAINERS



Re-adapted from @swardley (Simon Wardley)

DEMO #1
POWER OF CNAs
[AKA CNP DO-IT-YOURSELF]

AN APP STORY... IN MICROSERVICES



A **Honeypot** is a computer security mechanism set to detect, deflect, or, in some manner, counteract attempts at unauthorized use of information systems

Honeypots Provisioning

Honeypot #1 (honeypot.py)

Honeypot #2 (honeypot.py)

Honeypot #N (honeypot.py)

Receiver (receiver.py)

Persistency

AN APP STORY… OVER MANY CLOUDS + TOOLS

Visualization Layer

Service Discovery

etcd

etcd

Infra as Code

Honeypots Provisioning

amazon web services™

Google Cloud Platform

Receiver
(receiver.py)

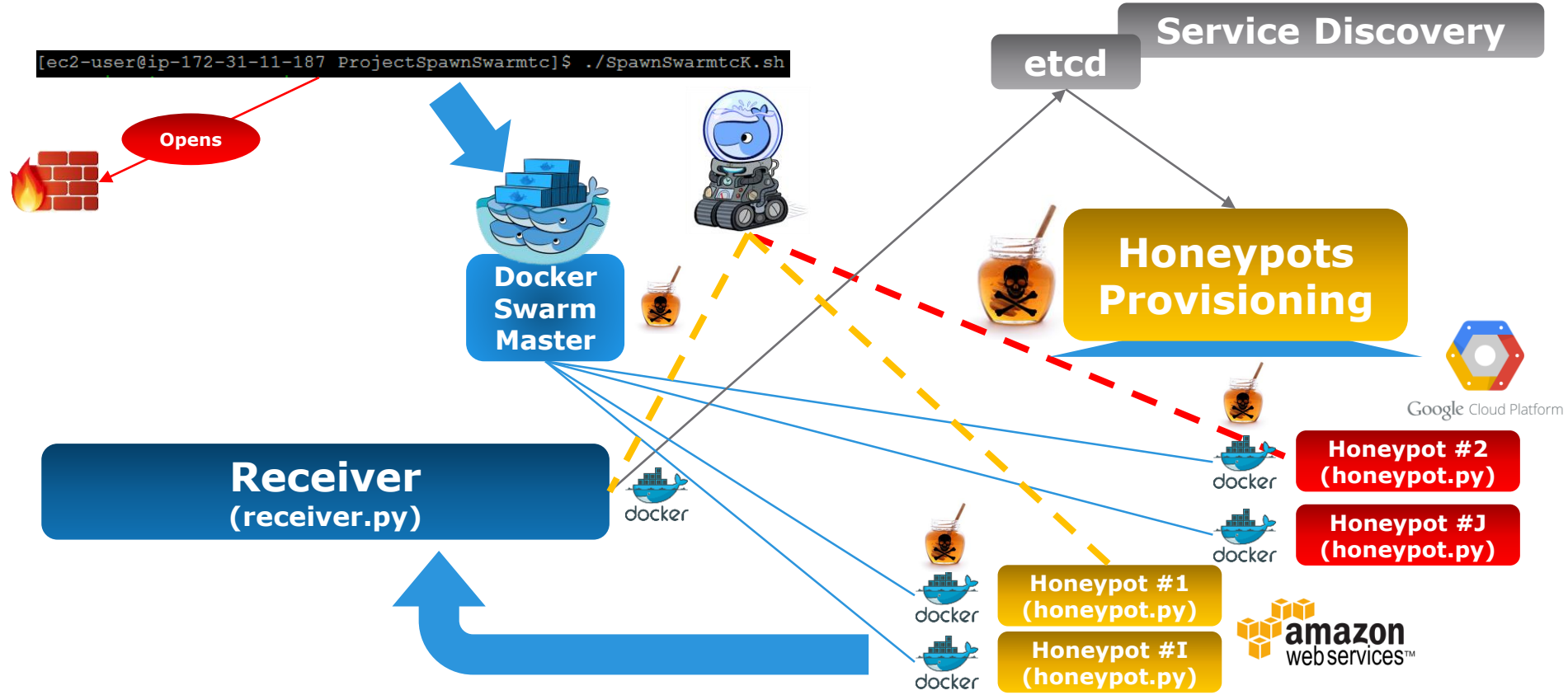
Agility

Honeypot #1 (honeypot.py)

Honeypot #2 (honeypot.py)

Honeypot #N (honeypot.py)

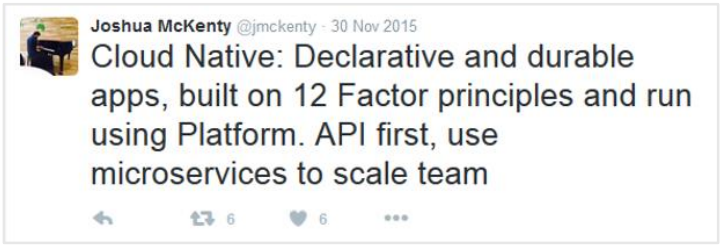
AN APP STORY... WITH MANY TOOLS

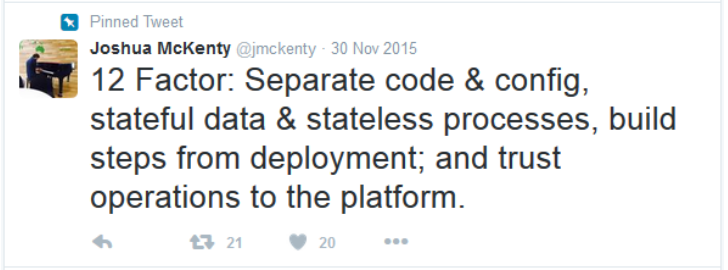


IN OTHER WORDS…

GOAL: Continuous Delivery
WHAT: Cloud Native Apps
HOW: Cloud Native Platform
THROUGH: DevOps Culture

IN OTHER TWEETS





HOW CAN YOU APPROACH IT?

TWO TYPES OF CLOUD NATIVE PLATFORMS

ASSEMBLED	PRESCRIPTIVE
UNSTRUCTURED	STRUCTURED
Ultra-Flexible	Standardized
Multiple Configurations	Built-in Availability and Security
Platform = Core Competency	Automate & Simplify Operations
Few Applications	Many Applications
DIY-focused	Buy-focused

TWO TYPES OF CLOUD NATIVE PLATFORMS

ASSEMBLED	PRESCRIPTIVE
UNSTRUCTURED	STRUCTURED

MESOS

kubernetes

CLOUDFOUNDRY

EMC²

TWO TYPES OF CLOUD NATIVE PLATFORMS

ASSEMBLED	PRESCRIPTIVE
UNSTRUCTURED	STRUCTURED
FEW	MANY
How many apps in the enterprise?	
YES	NO
Is the platform a core competency?	

Platform = C Competency

Au ns

kubernetes

EMC²

TWO TYPES OF CLOUD NATIVE PLATFORMS

ASSEMBLED	PRESCRIPTIVE
UNSTRUCTURED	STRUCTURED

EMC²

TWO TYPES OF INFRASTRUCTURE TO BUILD ON

BUILD IT YOURSELF

HYPER-CONVERGED

YES

Is racking, stacking, **and automating** a core competency?

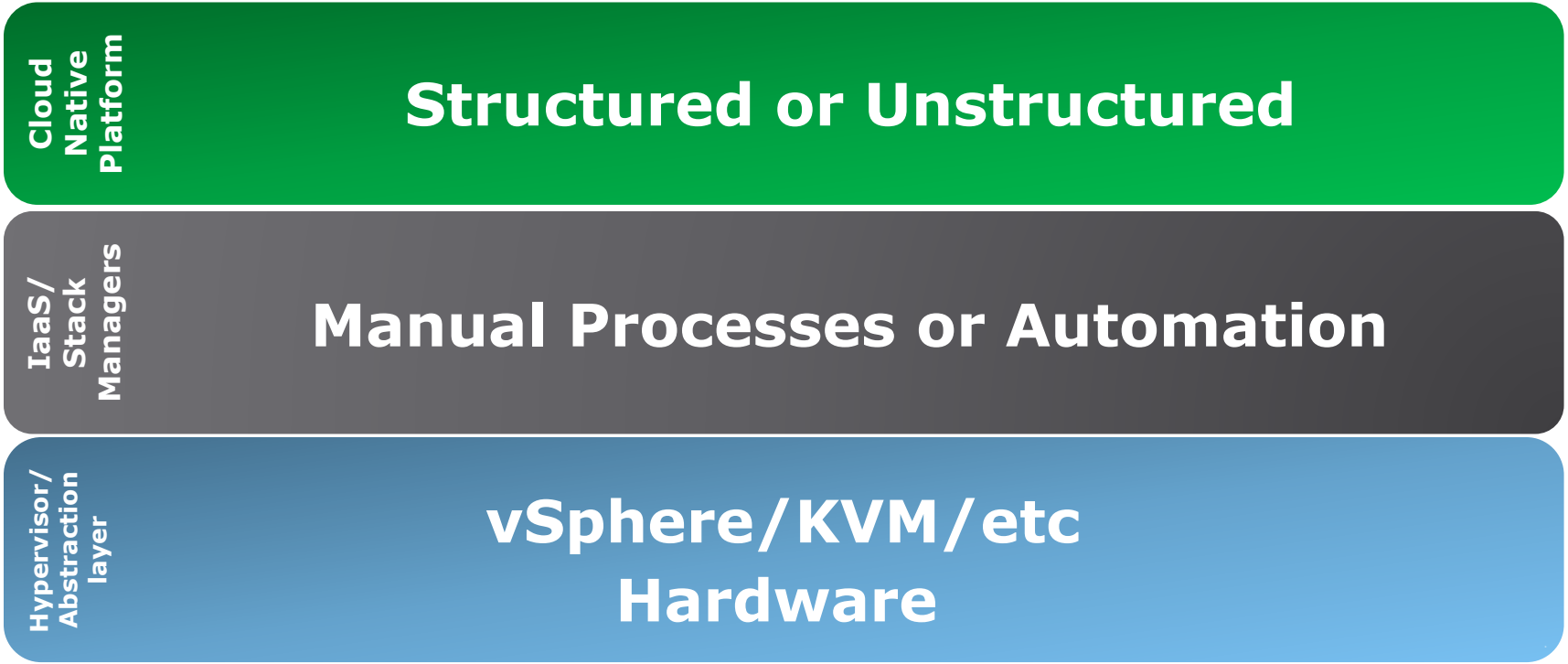
NO

LET’S SEE HOW A STRUCTURED PLATFORM LOOKS LIKE

DEMO #2
THE STRUCTURED EXPERIENCE
[AKA cf push tc]

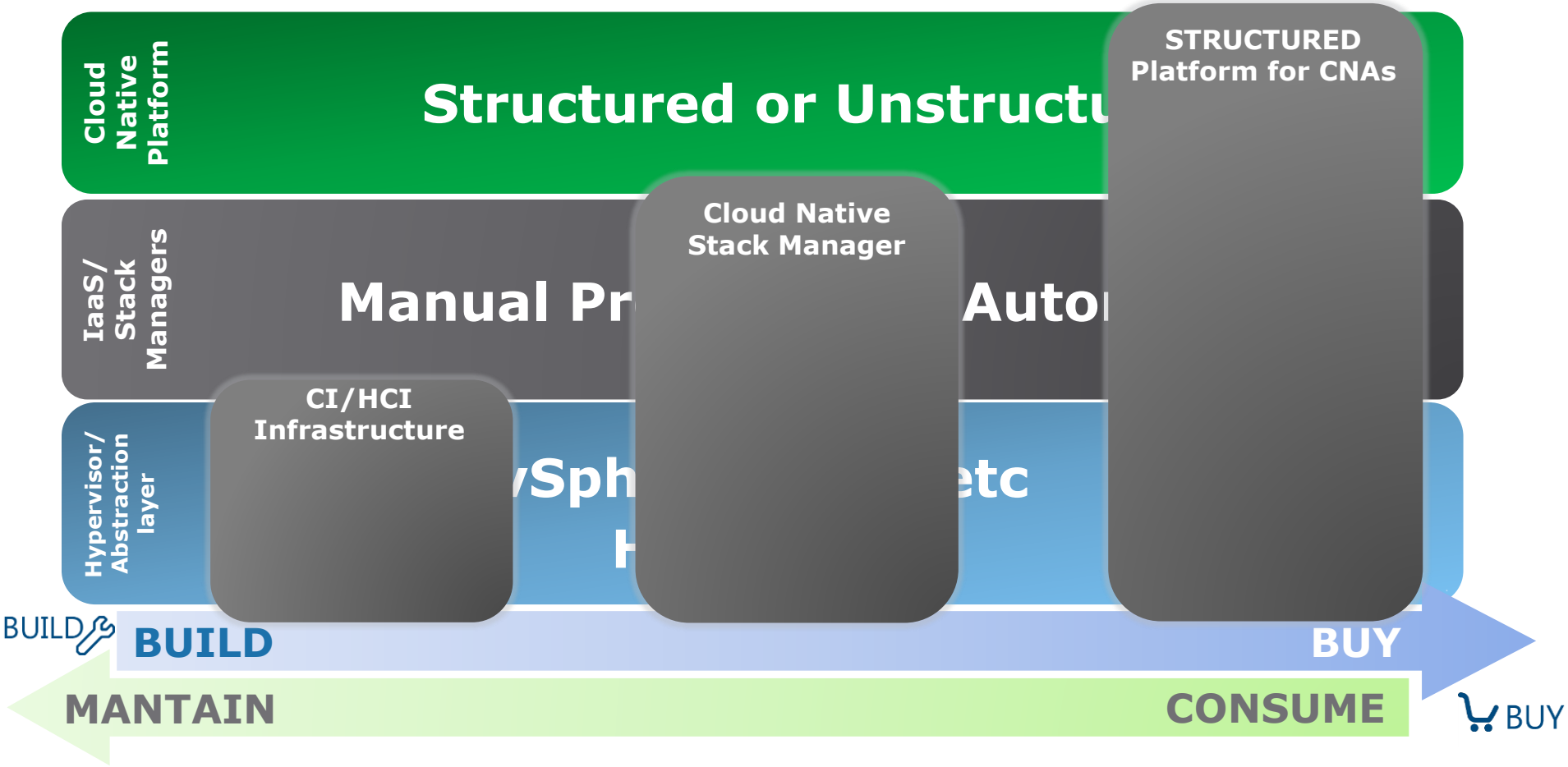
IN ESSENCE..
WHICH ARE THE LAYERS IN THIS NEW WORLD?

LAYERS FOR CLOUD NATIVE APPLICATIONS



*DIY= Do IT Yourself

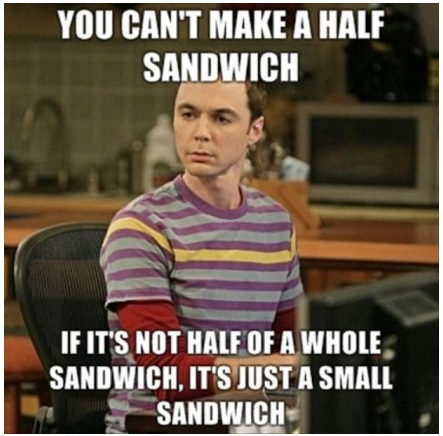
HOW TO SIMPLIFY CNAS DEPLOYMENT



*DIY= Do IT Yourself

RECAP / CLOSING COMMENTS

- Clear and Solid **Business Needs for Cloud Native Apps**
- Many Tech/options to choose from
- **Structured approach** offers many **advantages**
- **Purpose built** and Simplified **Infrastructure** can definitely help



THANK YOU!!!