Total virtualization

Before we begin

- ✓ Your aims and expectations

Virtualization

- ✓ Why do we need virtualization?
- ✓ Cloud stack technologies
- ✓ Virtual future
- ✓ Future of our course

Providers of virtualization and cloud solutions

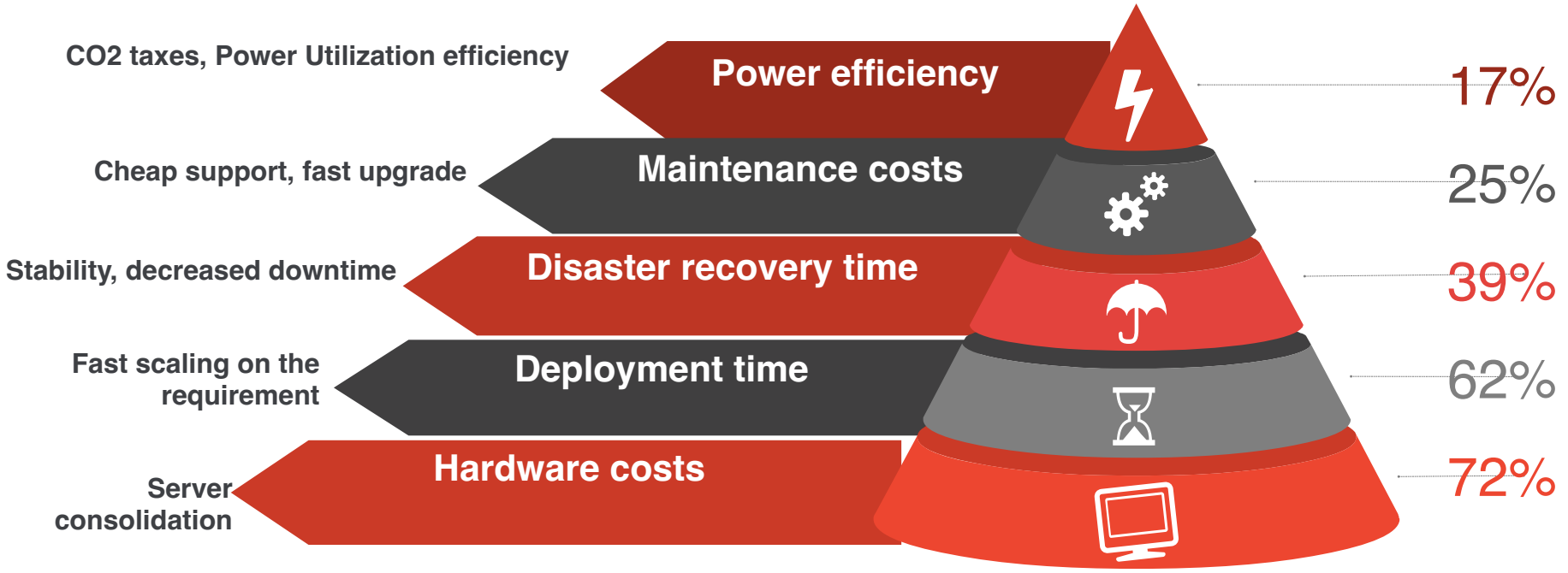
ARM®
amazon
intel®
FreeBSD
Microsoft®
IBM
EMC²
CITRIX®
AMD
Parallels™
ORACLE®
Linux™
Google™

But why businesses needs virtualization?

But why businesses needs virtualization?

ROI

But why business needs virtualization?



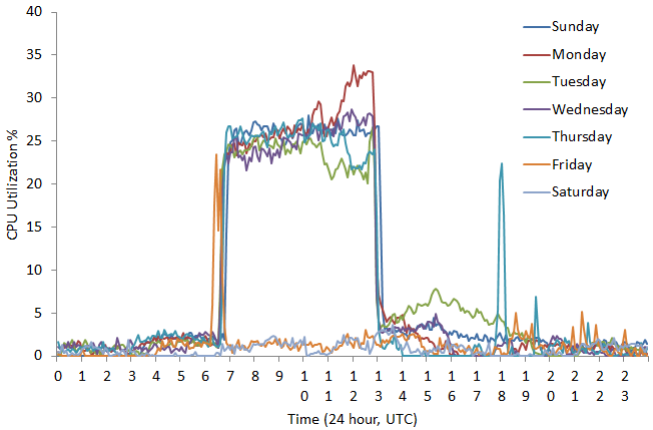
Development costs are reduced as well: faster testing, faster installations

(c) Rightscale

Server consolidation

Time-sharing apartments

Server utilization



(c) Huan-Liu

Utilization level< 12%

(c) Gartner

Virtualization couldn’t be sold

Virtualization couldn’t be sold

But cloud and infrastructure can

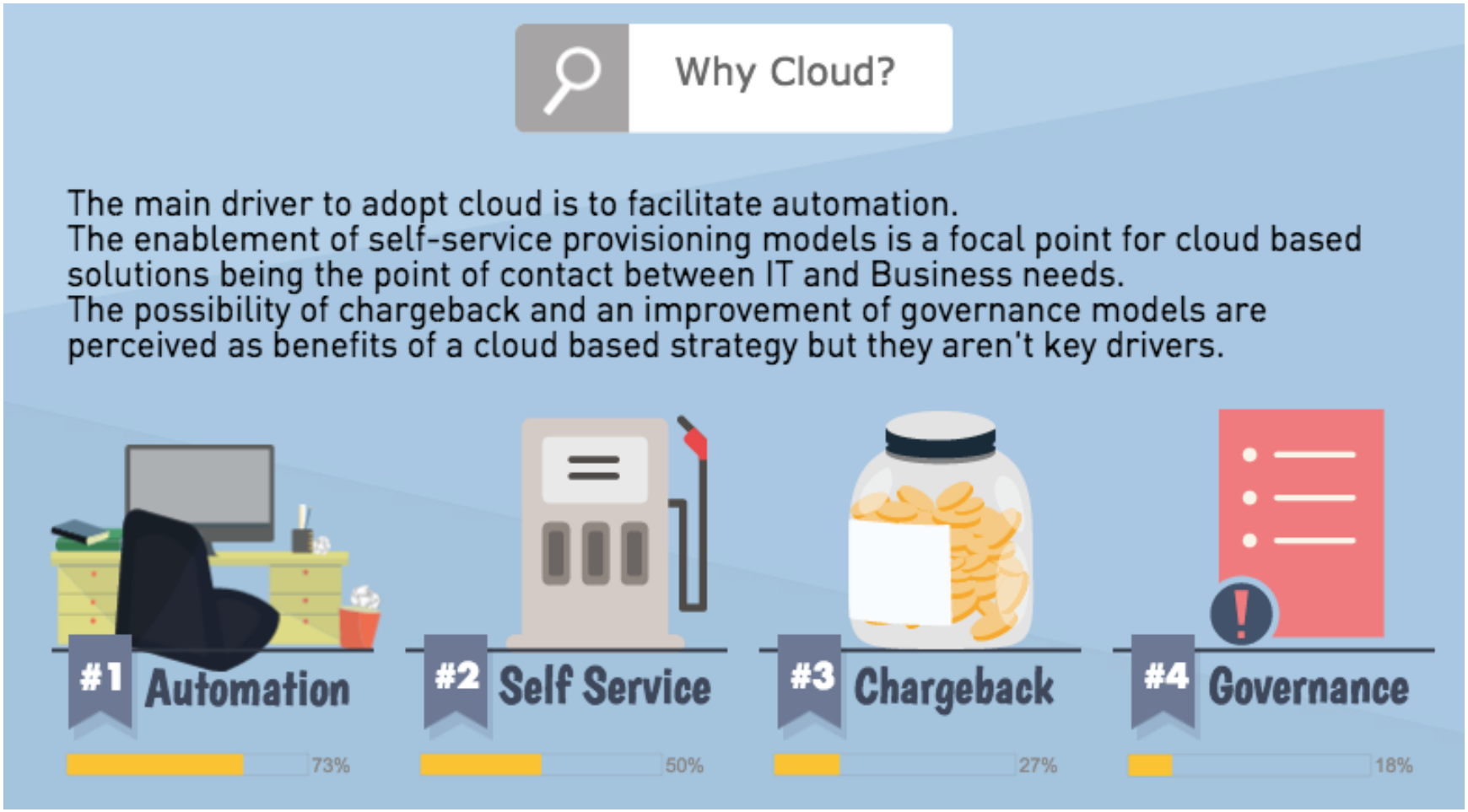
Cloud

Cloud Computing is a model for enabling 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

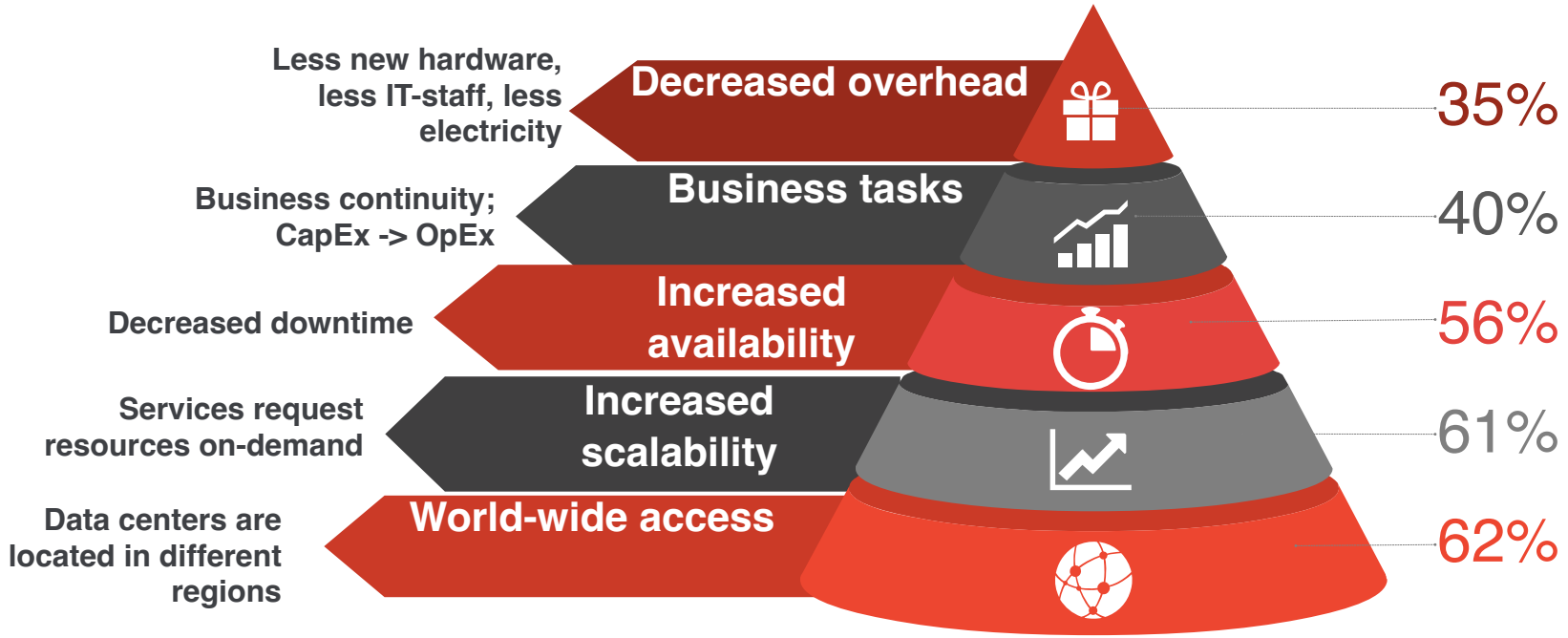
Cloud

- ✓ On-demand self-service. The ability for an end user to sign up and receive services without the long delays that have characterized traditional IT.
- ✓ Broad network access. Ability to access the service via standard platforms (desktop, laptop, mobile etc).
- ✓ Resource pooling. Resources are pooled across multiple customers
- ✓ Rapid elasticity. Capability can scale to cope with demand peaks
- ✓ Measured service. Billing is metered and delivered as a utility service

Cloud. Why?


Why Cloud?
The main driver to adopt cloud is to facilitate automation.
The enablement of self-service provisioning models is a focal point for cloud based solutions being the point of contact between IT and Business needs.
The possibility of chargeback and an improvement of governance models are perceived as benefits of a cloud based strategy but they aren't key drivers.
#1 Automation
73%
#2 Self Service
50%
#3 Chargeback
27%
#4 Governance
18%

Cloud. Why? (2)

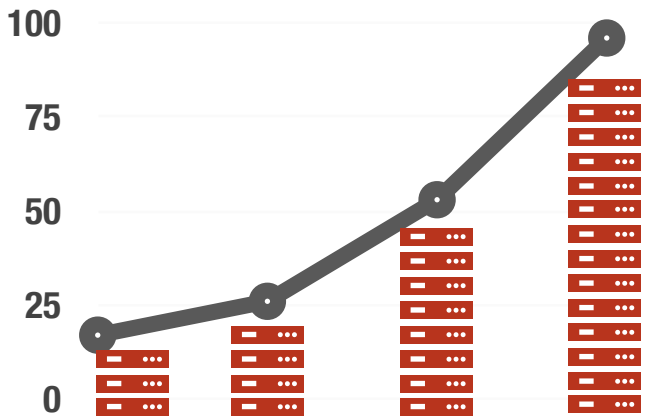


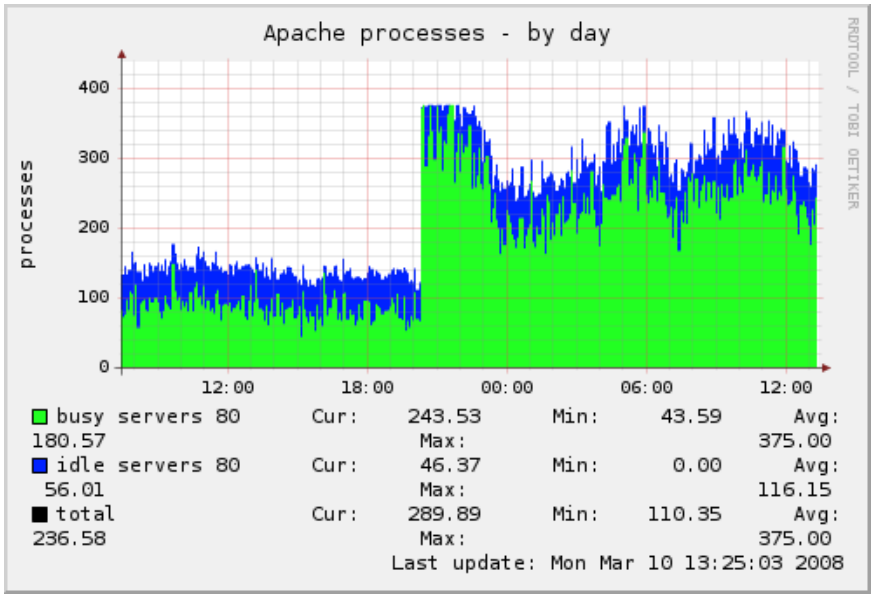
(c) RightScale

Scalability (1)

Definition

The ability of system to process the increased load with the help of increased resources



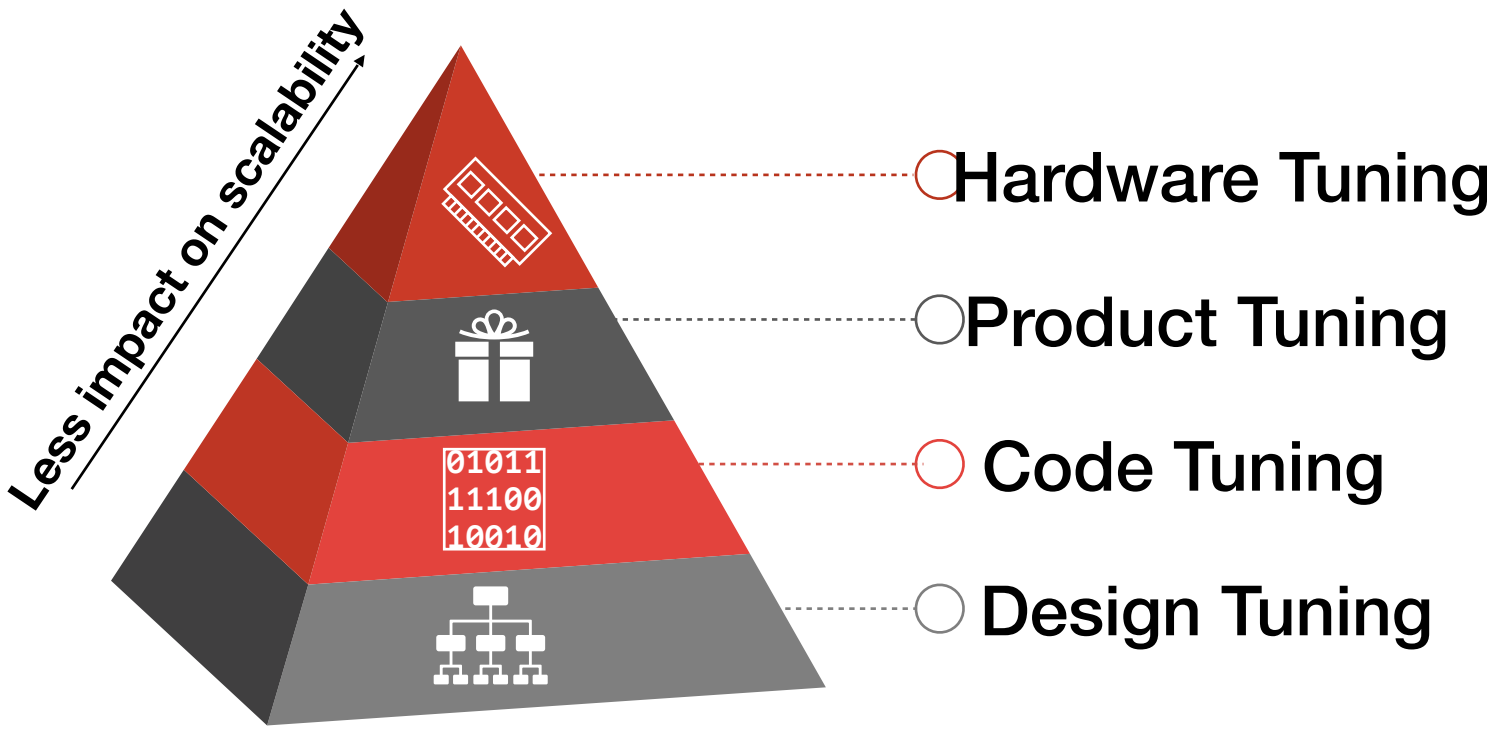


Scalability (2)

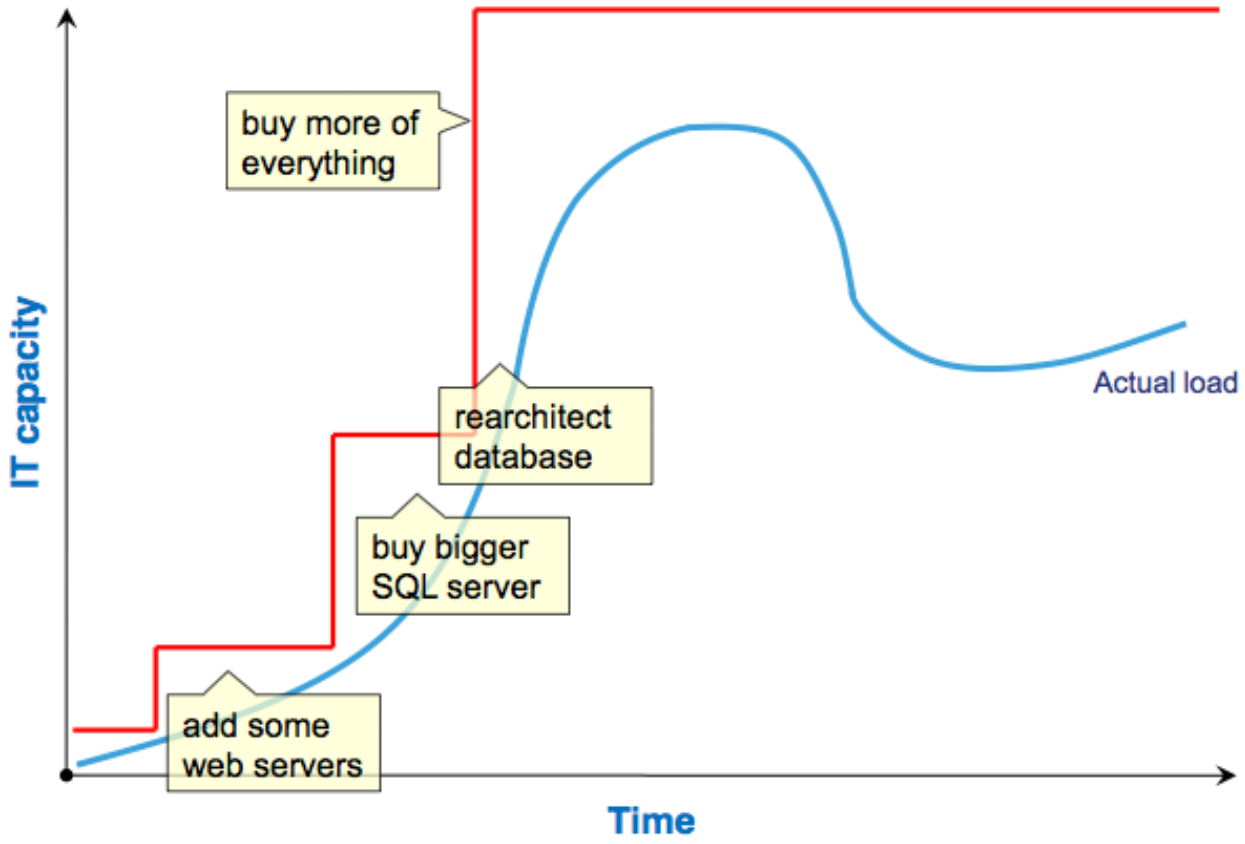
An always-on service is said to be scalable if adding resources to facilitate redundancy does not result in a loss of performance.

Werner Vogels,
Amazon CTO

Scalability (3)

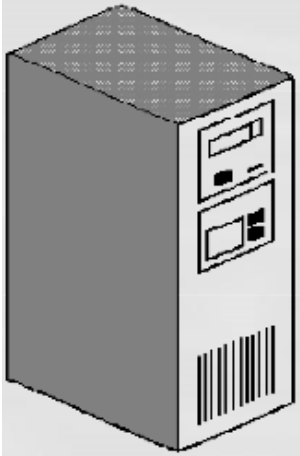


Scalability (4)



Scale Up vs Scale Out

Option 1: Scale UP



Get a bigger computer.

If one ox could not do the job they did not try to grow a bigger ox, but used two oxen. When we need greater computer power, the answer is not to get a bigger computer, but...to build systems of computers and operate them in parallel.

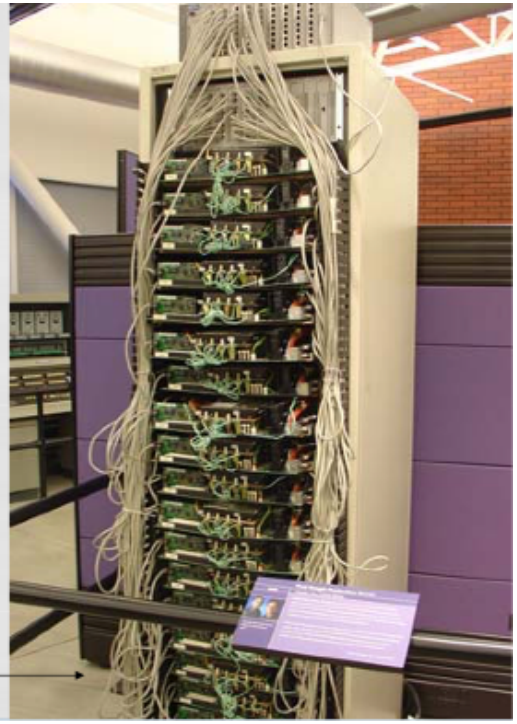
(Grace Hopper)

ixquotes.com

Option 2: Scale Out

Get more computers.

Google Hardware c. 1999 →



Scale Up vs Scale Out(2)

Scale Out: Files

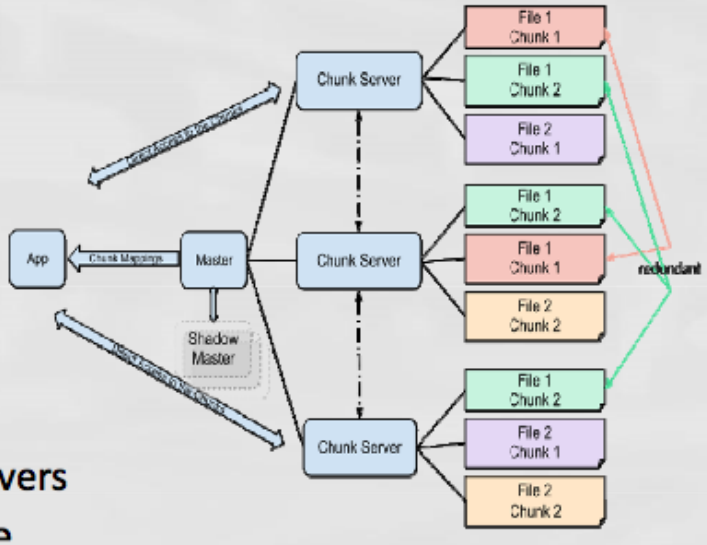
Google's Problem:

- Search the entire Internet
- Instantly

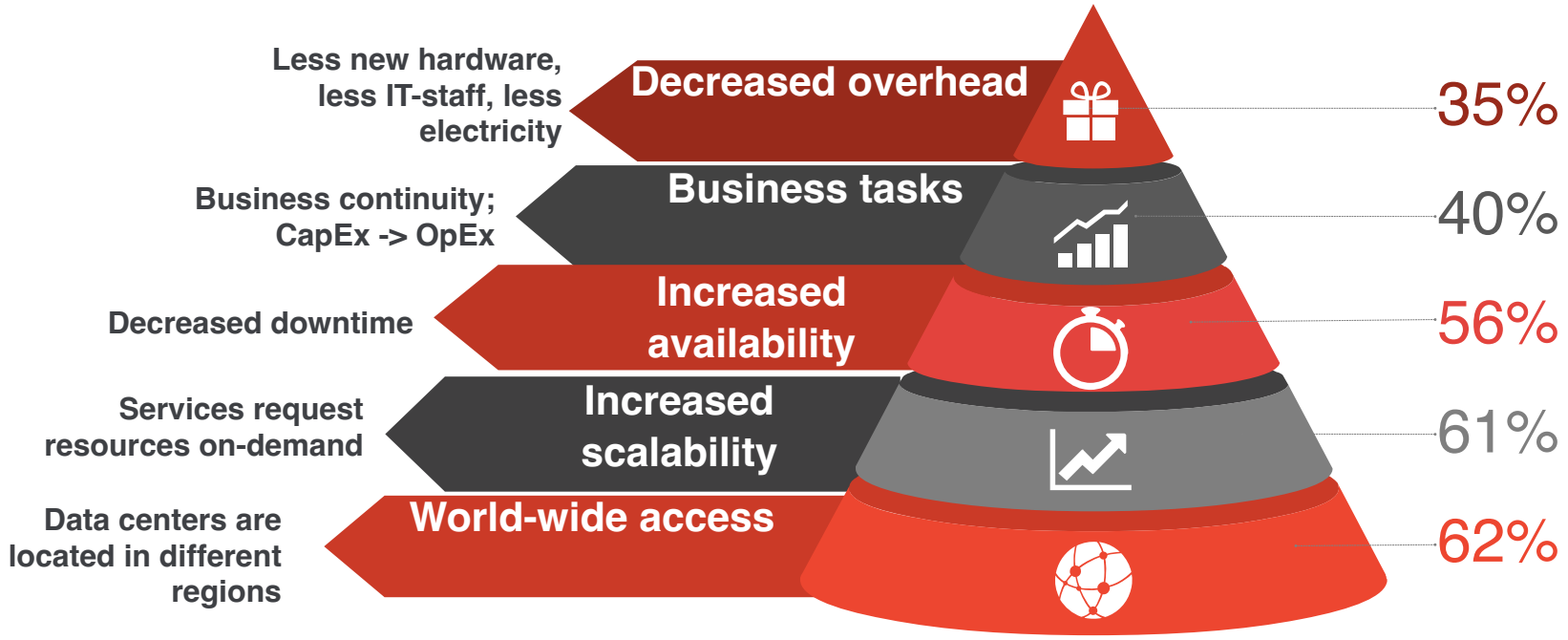
Response:

Horizontal Scaling

- Break files into small pieces
  - Make three copies of each piece
  - Spread the pieces across many servers
- Manage the pieces with software

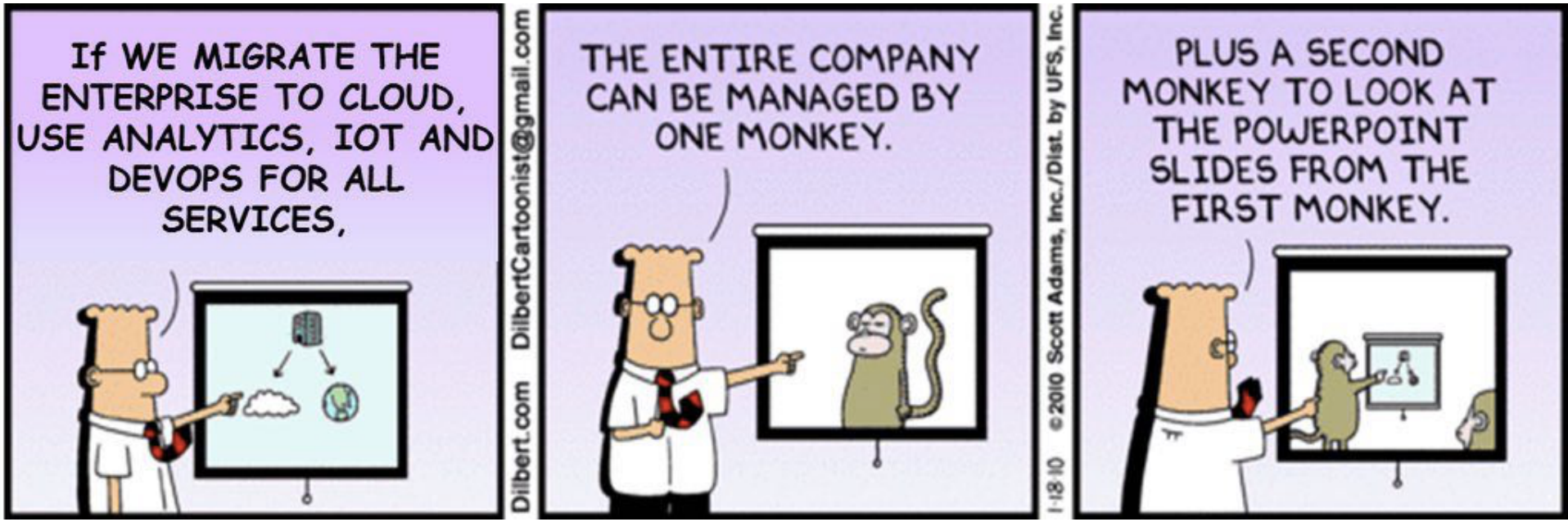


Cloud. Why? (2)



(c) RightScale

Cloud. Why? (3)


If WE MIGRATE THE ENTERPRISE TO CLOUD, USE ANALYTICS, IOT AND DEVOPS FOR ALL SERVICES,
THE ENTIRE COMPANY CAN BE MANAGED BY ONE MONKEY.
PLUS A SECOND MONKEY TO LOOK AT THE POWERPOINT SLIDES FROM THE FIRST MONKEY.
Dilbert.com DilbertCartoonist@gmail.com
1-18-10 ©2010 Scott Adams, Inc./Dist. by UFS, Inc.

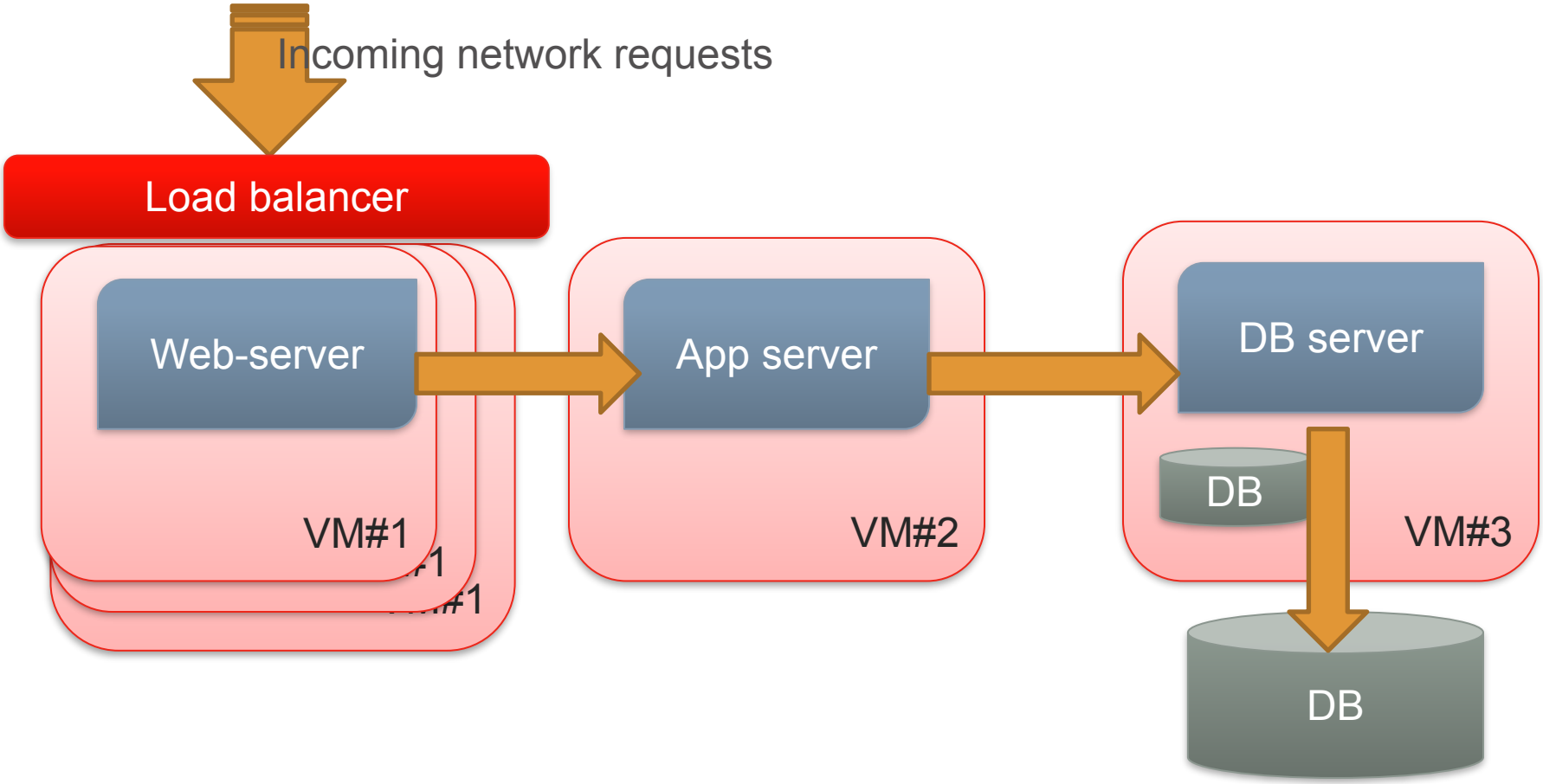
Cloud levels

Infrastructure
User buys virtual hardware

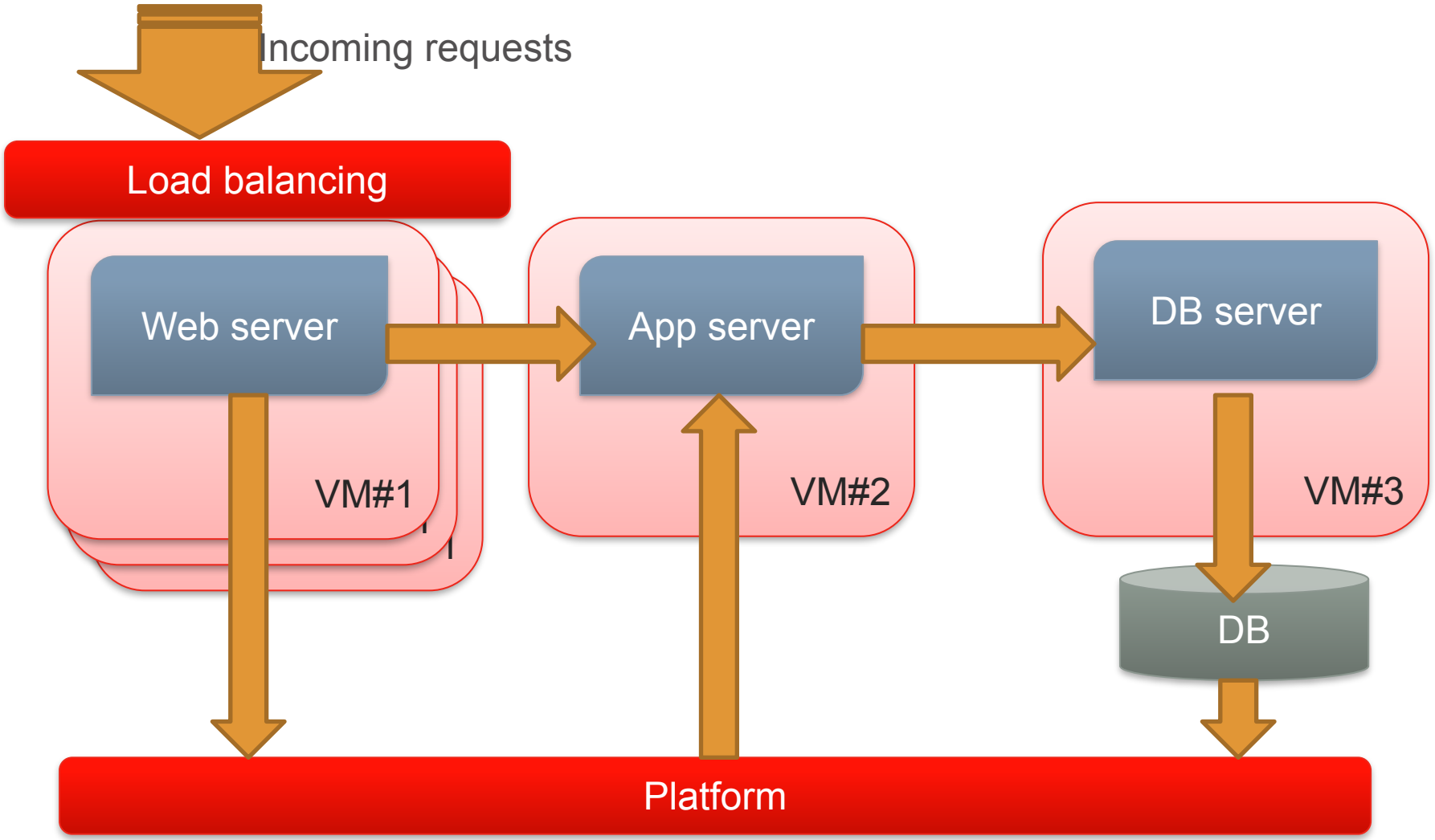
Platform
User buys access to the platform that controls virtual hardware

Service
User buys access to the app that works on the platform that controls virtual hardware

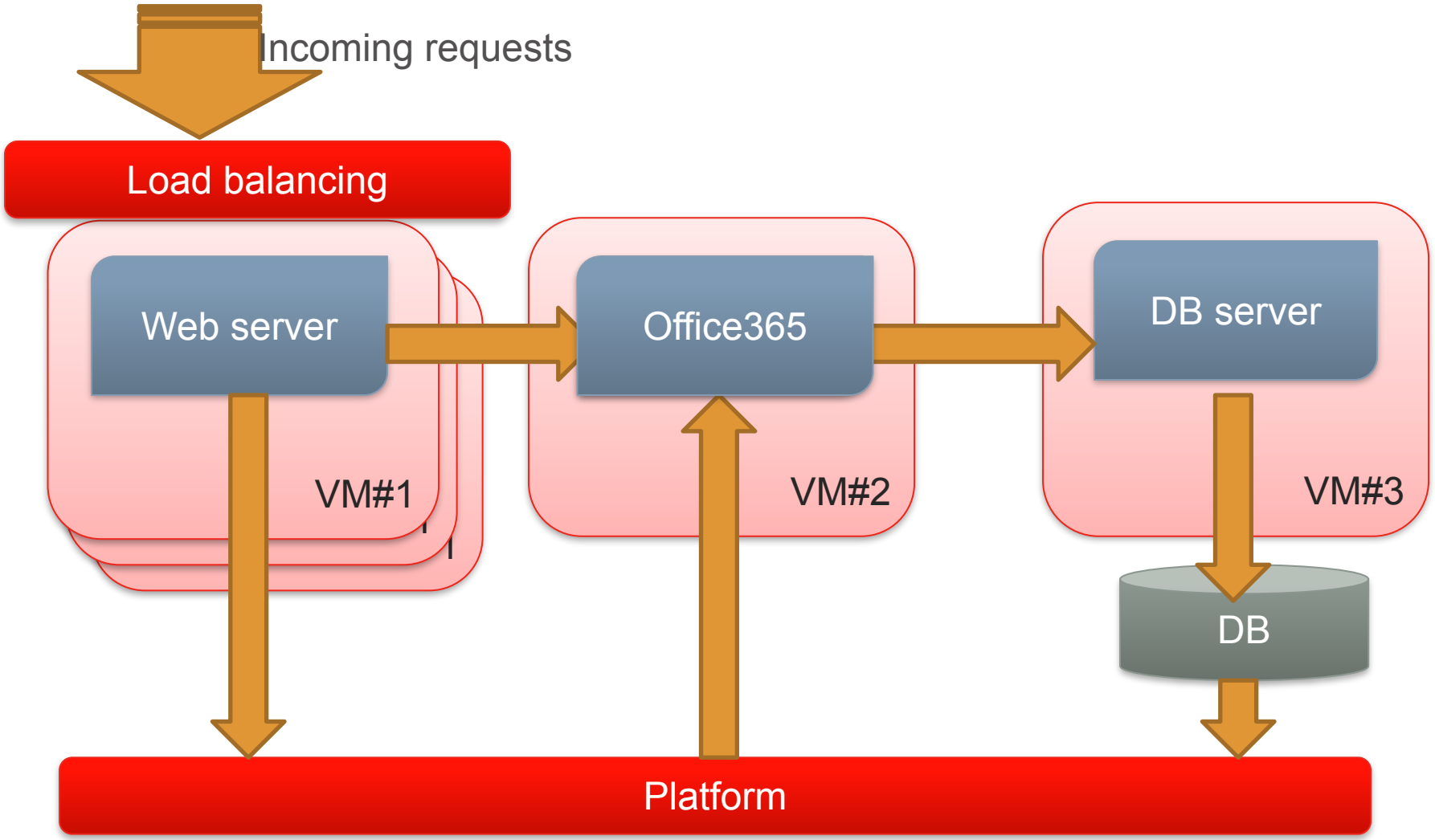
Clouds: IaaS -> PaaS



Clouds: IaaS -> PaaS



Clouds: PaaS -> SaaS



Clouds: service levels

Software as a Service (SaaS).

Cap = Use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface or a program interface. The consumer does not manage or control the underlying cloud infrastructure with the possible exception of limited user-specific application configuration settings.

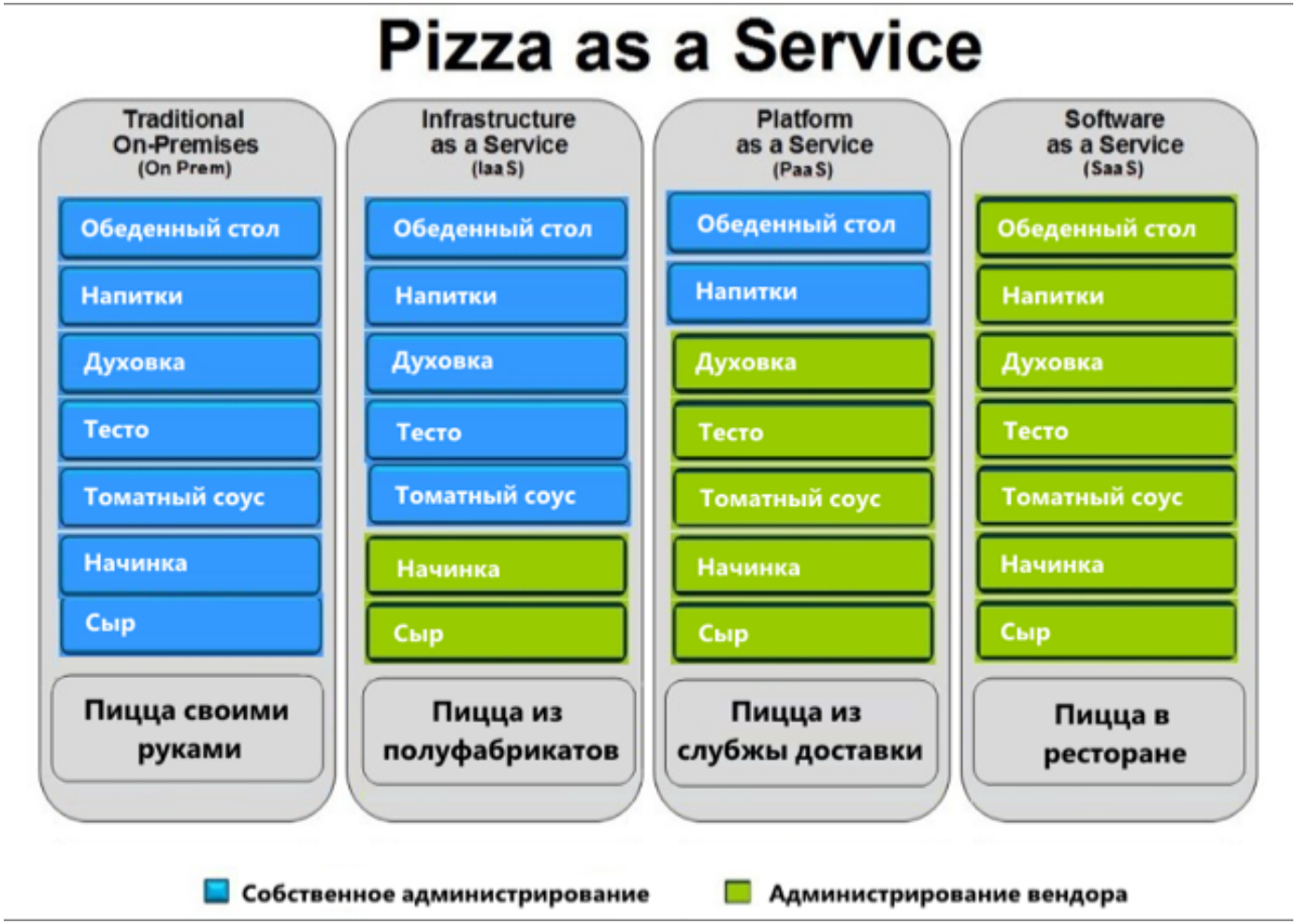
Platform as a Service (PaaS).

Cap = Deploy onto the cloud infrastructure consumer-created or acquired apps. The consumer does not manage or control the underlying cloud infrastructure, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

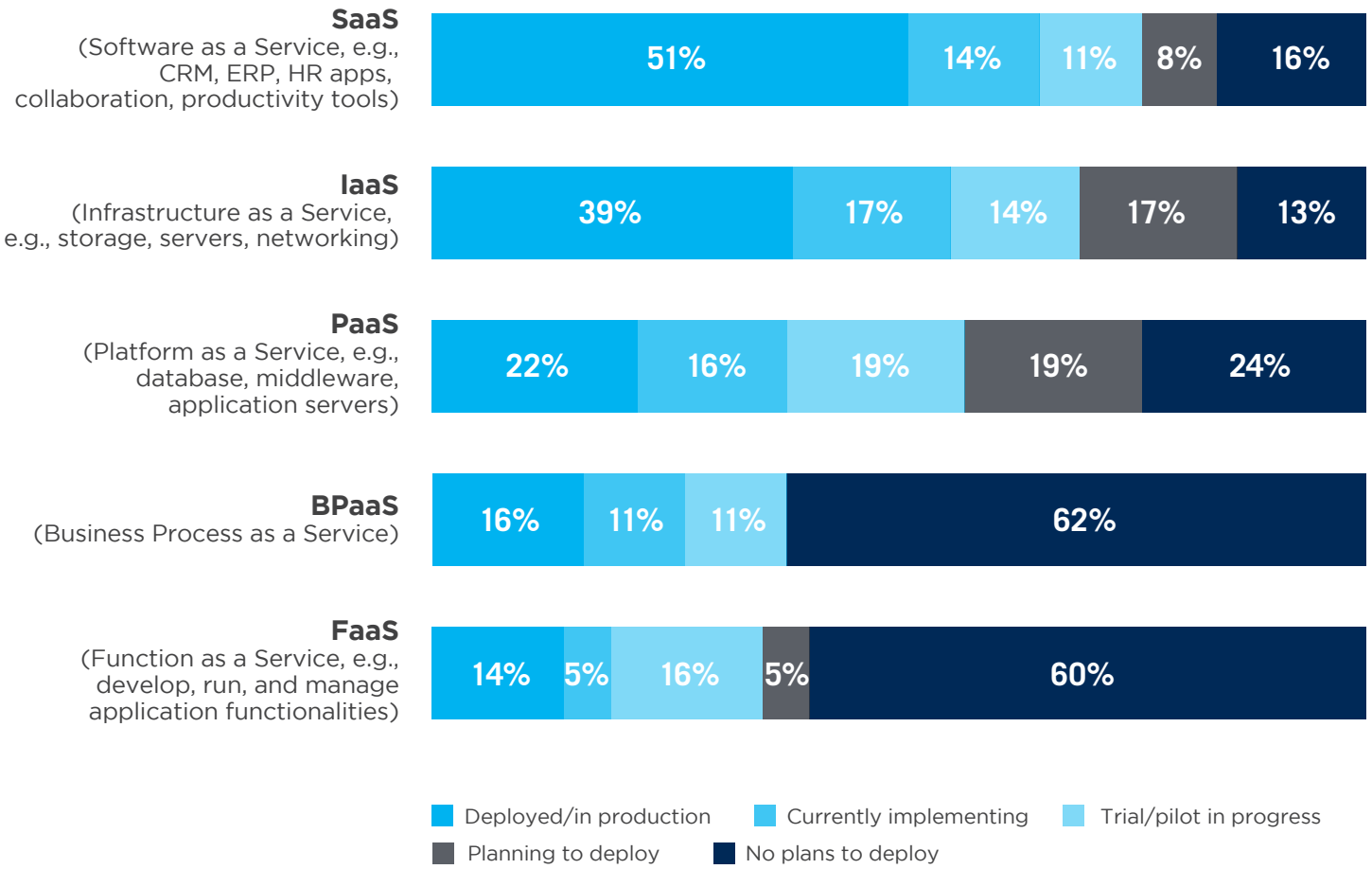
Infrastructure as a Service (IaaS).

Cap = Provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include OSs and apps. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls).

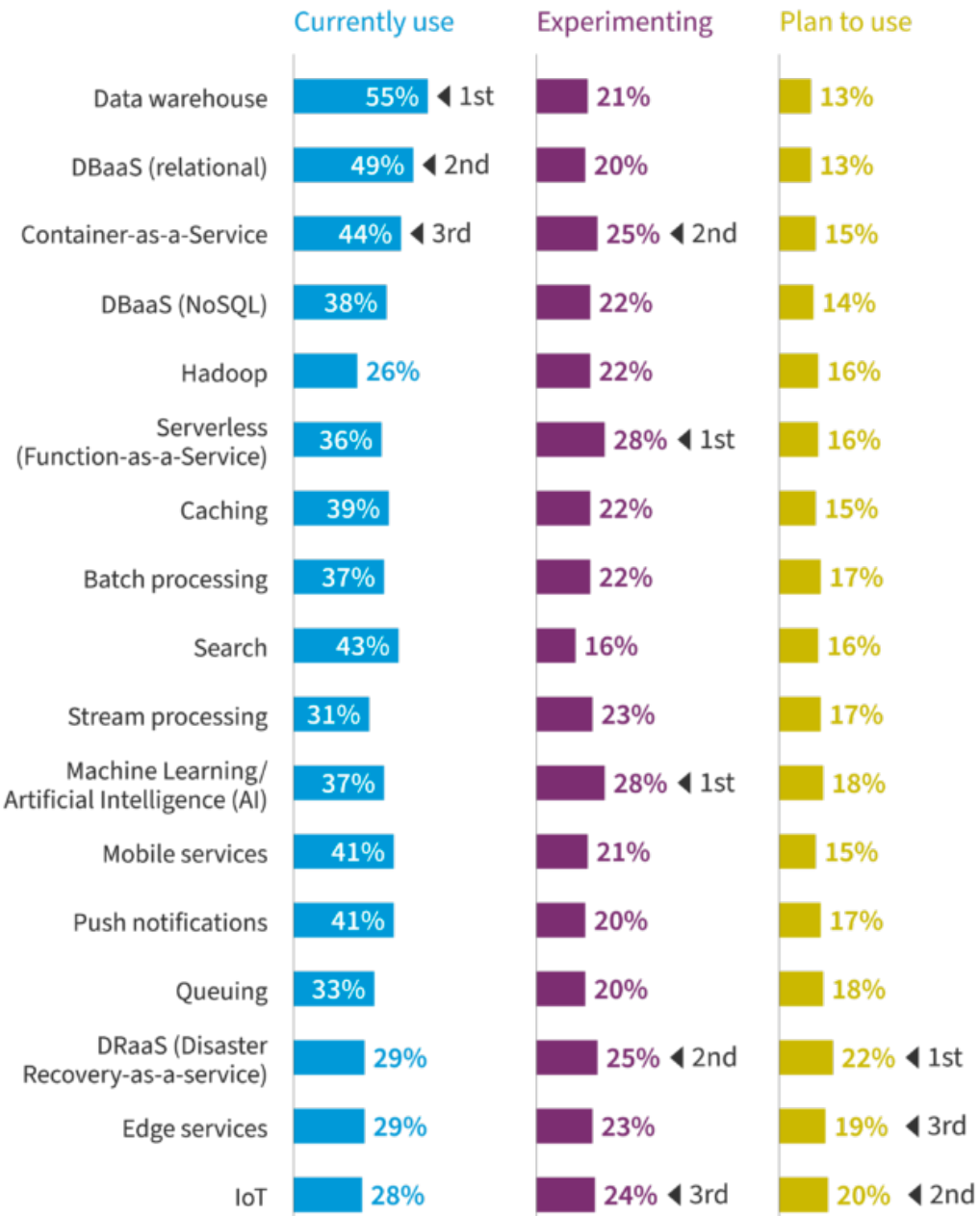
Service layers in pizza



Cloud adoption



Adoption



N=753
Source: Flexera 2022 State of the Cloud Report

Trends

- Container as a Service
- AI as a Service (ML as a Service)
- DRaaS
- Edge Services
- Function as a Service

Function as a Service

an event-driven computing service that runs code in response to events and automatically manages the computing resources required by that code.

Amazon Lambda, Google Function, Azure Functions

Business Process as a Service

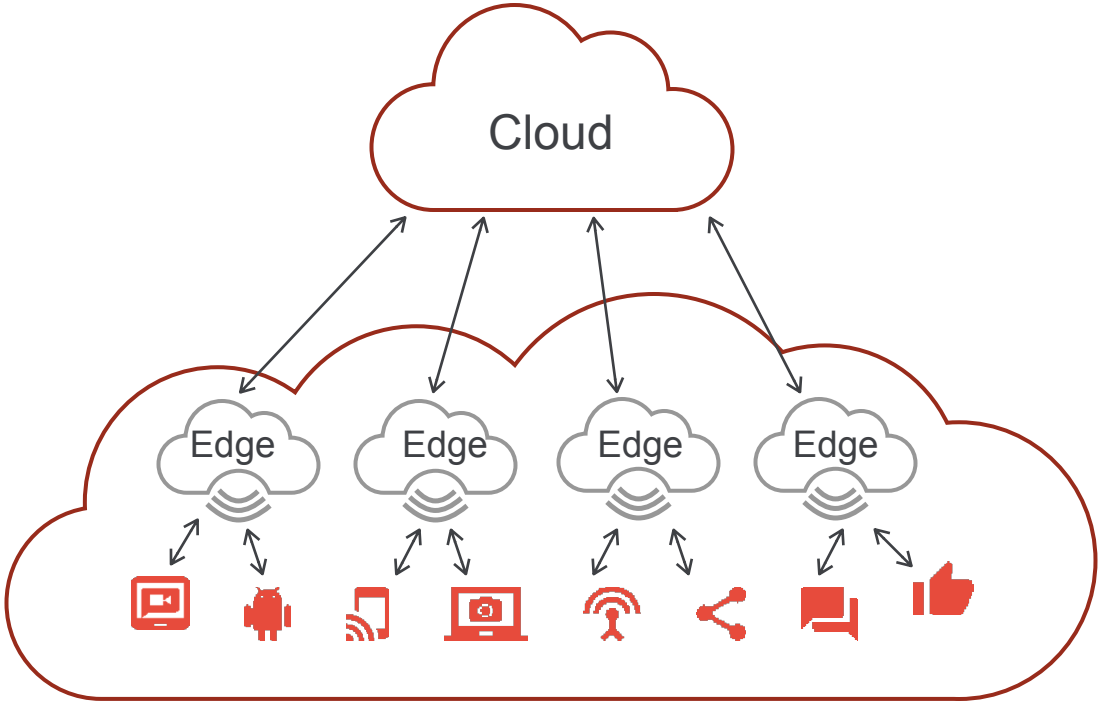
Disaster Recovery As a Service

- ✓ Autobackup
- ✓ Quick recovery
- ✓ Data on-premise goes

Zerto

Containers as a Service

Fog computing (Edge)



Boost of operating systems

Unikernel

- Run inside
- Only in VM
- Drop all security and isolation, drop resource management
- Presupposition: run only one process
- OSV, HermitCore, MirageOS

Minimalistic OS

- Run outside
- On the host or (mostly) in VM
- Drop everything outside virtualization and process isolation
- Presupposition: run nothing but VE(VM or CT)
- RancherOS, AtomicHost, CoreOS, Amazon FireCracker, Bottlerocket

Virtualization future

Future of our course

Virtualization course

Virtualization deep dive

- Virtualization theory
- Algorithms, methodology, data structures
- Intel architecture basics, OS intro
- For architects and system programmers

Cloud deep dive

- Cloud service levels
- Labs on apps deployment
- Orchestration, API
- For CTOs, architects, DevOps

Future of our course: virtualization

- A virtualization introduction: basics, history. Reinvent the wheel
- CPU virtualization approaches: emulation, binary recompilation, stubbing, para-virtualization, hardware-assisted technologies
- Physical memory virtualization issues: compression, swap files, balloons
- Virtual paging, delivery of interrupts in virtual machines, time keeping in virtual machines
- Network!
- Virtualizing video
- Tools and migration

Future of our course: virtualization

- Introduction to containers
- OS-level virtualization architecture and OpenVZ project
- Docker
- Kubernetes

Future of our course: cloud

- DRS: not only resource distribution. Dynamic power management. Placement, planning horizon, etc.
- Failovers and backups
- PaaS development and microservices
- Migrating to clouds. Private and public clouds (choices, configuration issues)
- Cloud Storage
- Cloud and Virtualization security

Future of our course: remarks

- Each of your statement must be «proved» with the corresponding cite
- Not only the response, but your way to that response
- Plagiarism is evaluated as -10 for every student
- Labs may include «googling» and that’s intentional.

Thanks for your attention!