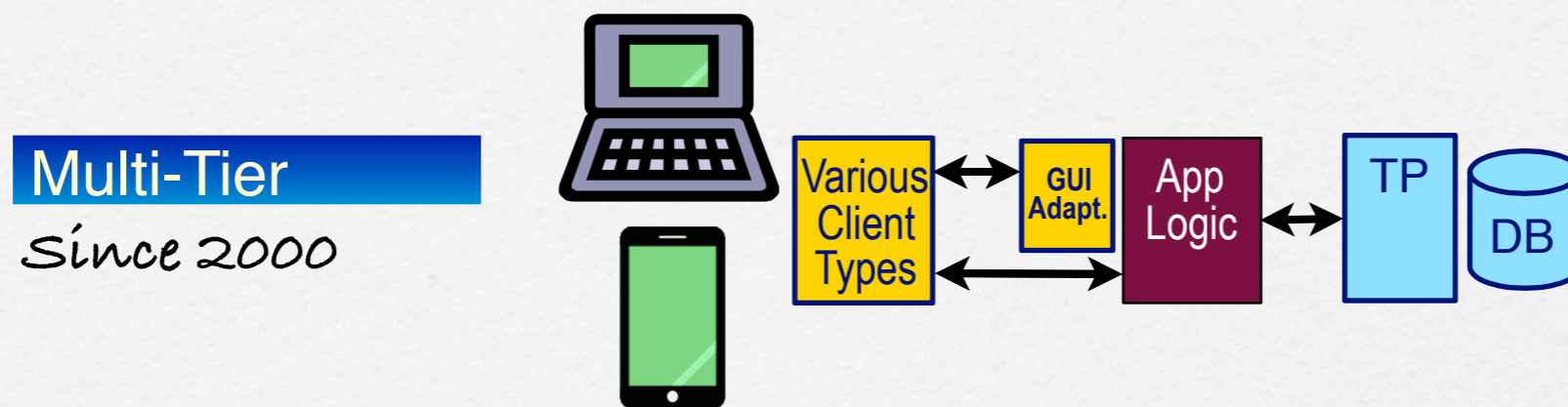


Multitiered Architectures & Cloud Services

Learning objectives

- ☐ Learn about enterprise computing
- ☐ Learn about multitiered architectures
- ☐ Learn about Java Enterprise Services
- ☐ Learn about cloud computing

Architecture evolution | Specialization



Enterprise computing

- Distributed Enterprise Applications exhibit critical needs:
 - highly available
 - highly reliable
 - highly scalable
 - highly secure
 - Etc.
- Software architects & developers must therefore be experts not only in the application domain, but also in these various orthogonal domains known as system qualities

Enterprise computing (2)

- ❑ In addition, with the advent of the web and of mobile communication, enterprise applications must now be able to interact via many devices on many channels
- ❑ Conclusion: software engineers must in addition aim at flexible, multi-channel & forward-looking distributed architectures

Application server

- ❑ Software that runs on some middle tier, between:
 - ❑ web-server (thin clients)
 - ❑ databases / legacy applications
- ❑ Support for clustering, load balancing, fail-over, connectivity to legacy systems, transaction processing, business logic, etc...
- ❑ Hosting environment for server-side components

Java Enterprise Services

- ❑ A set of standard APIs providing access to existing infrastructure services
- ❑ Enterprise Java APIs are platform & vendor neutral
- ❑ A business component model based on these APIs, i.e., that can be deployed on:
 - ❑ any hardware/operating system
 - ❑ any compliant applications server

→ The Java EE platform

Java EE | Overview

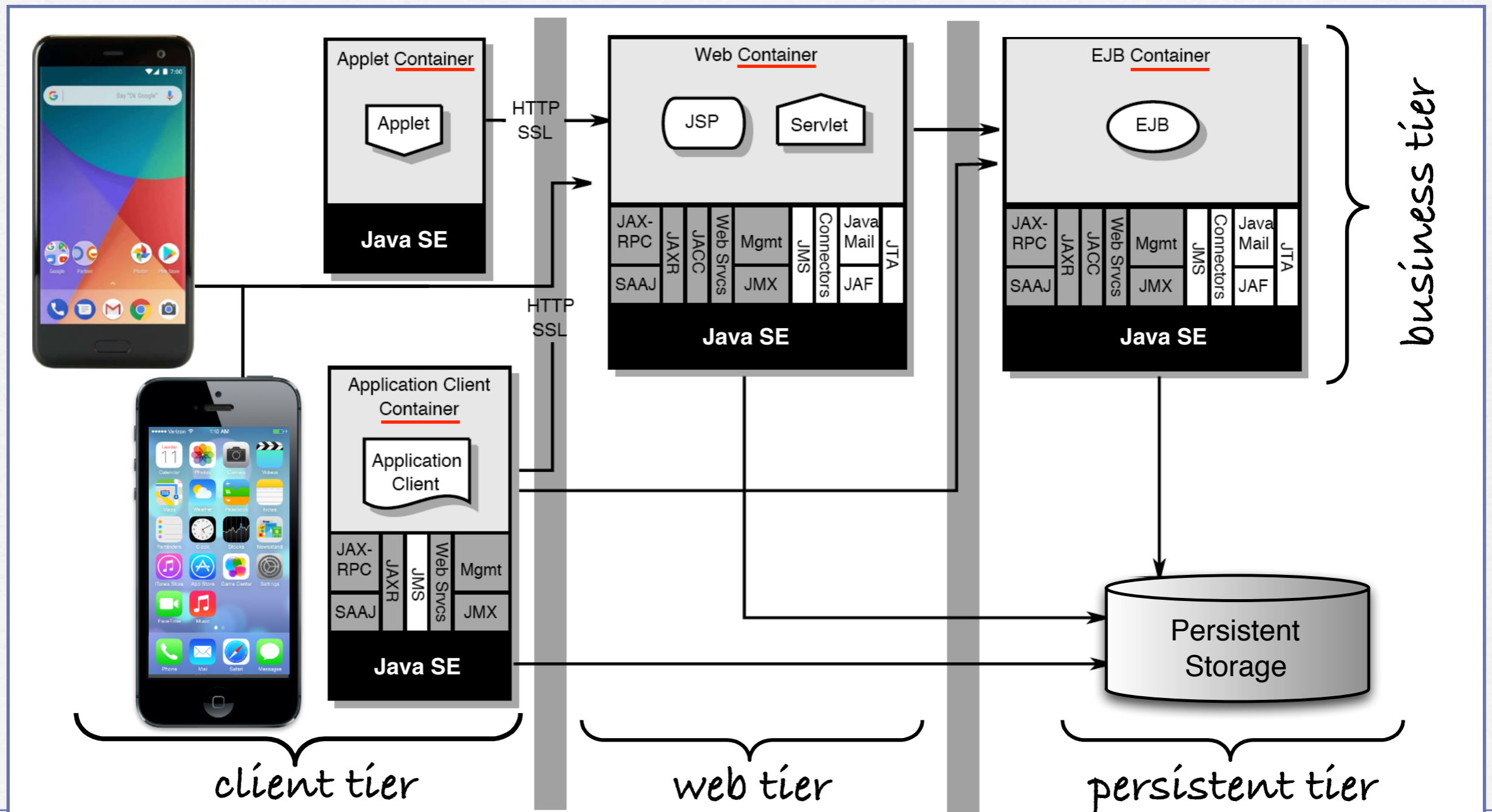
- Java EE stands for Java platform, Enterprise Edition
- Java EE is the specification of a distributed multitiered application model for enterprise applications, presented as a coherent set of programming APIs
- Implementations of the Java EE specification are usually proposed in the form of application servers



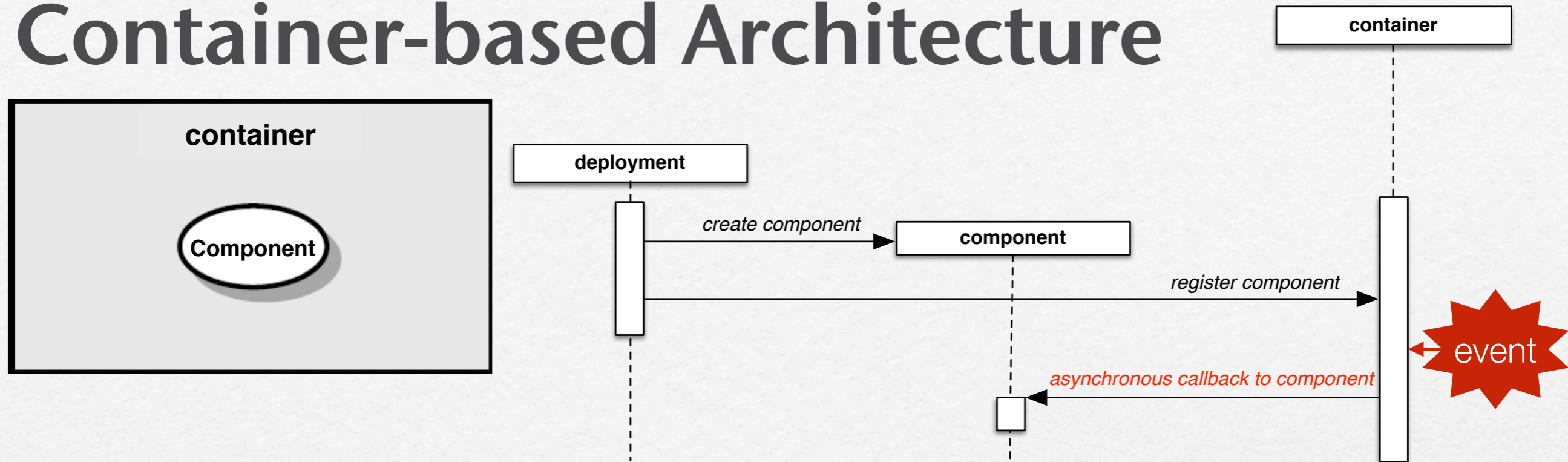
Enterprise Java APIs

- ❑ Distributed Objects: Web Services, Java RMI
- ❑ Object Directory: JNDI
- ❑ Database Access: JPA, JDBC
- ❑ Transactions: JTA, JTS
- ❑ Web: Servlets, JSP, tag libs
- ❑ Asynchronous interactions: Websockets, Futures
- ❑ Business Components Model: EJBs

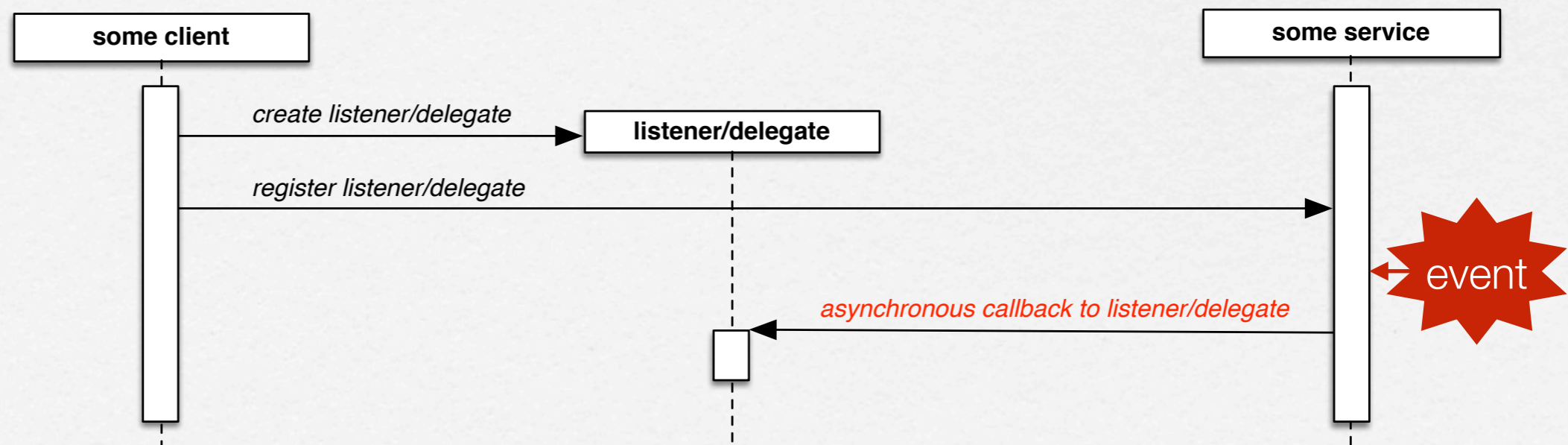
Container-based Architecture



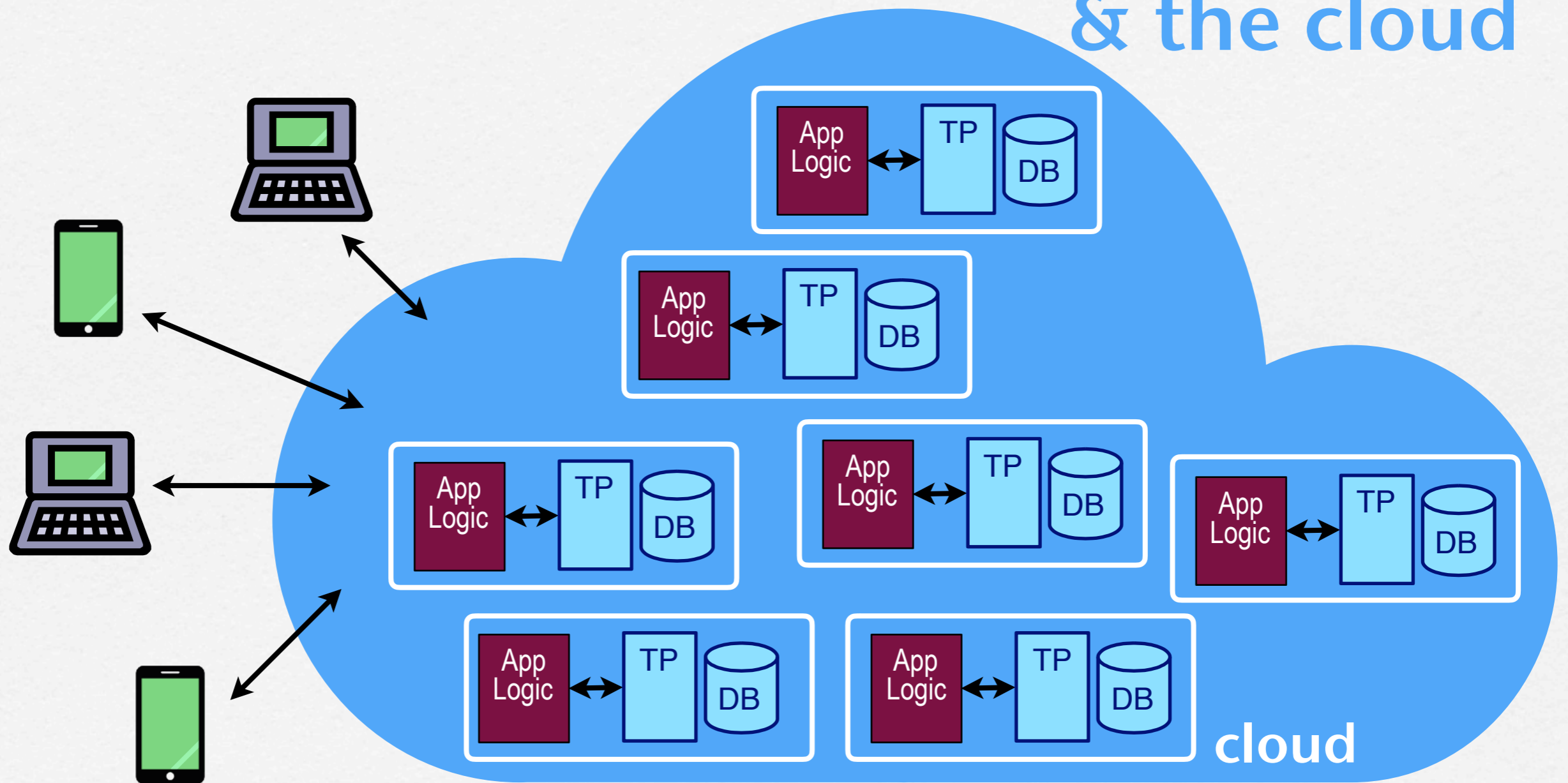
Container-based Architecture



The Generalized Asynchronous Callback Principle



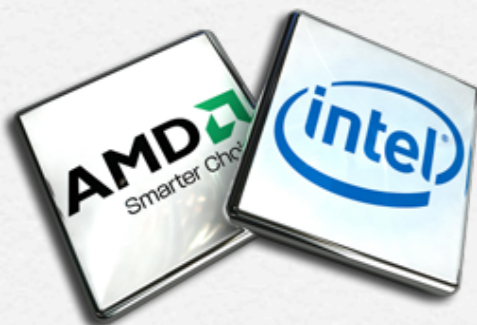
Architecture evolution | Virtualization & the cloud



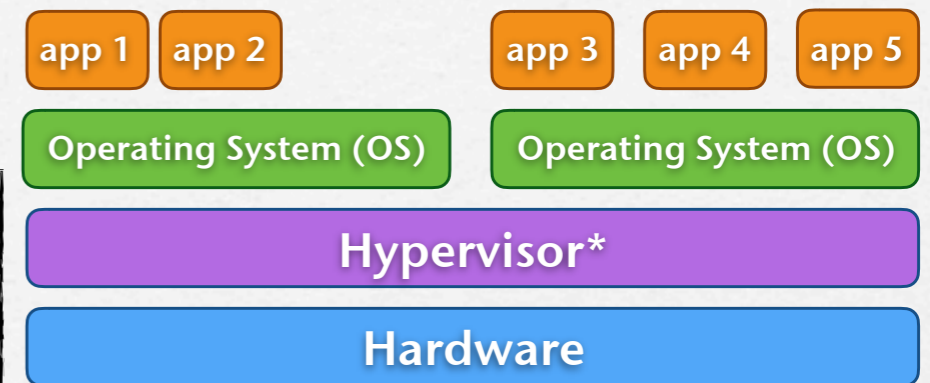
Virtualization | Principle



1960: first concept & implementation
in the **IBM S/360**



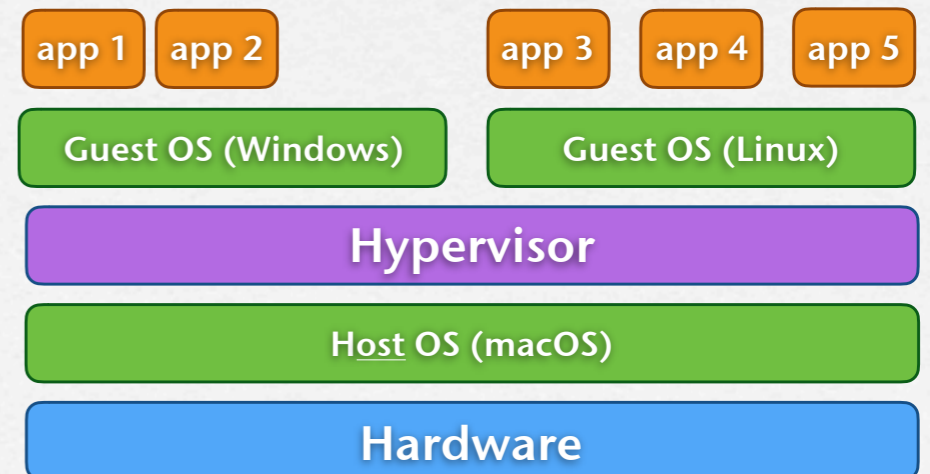
since 1999: rebirth of
this concept applying
it to the Intel processor architecture



*also called **Virtual Machine Monitor**



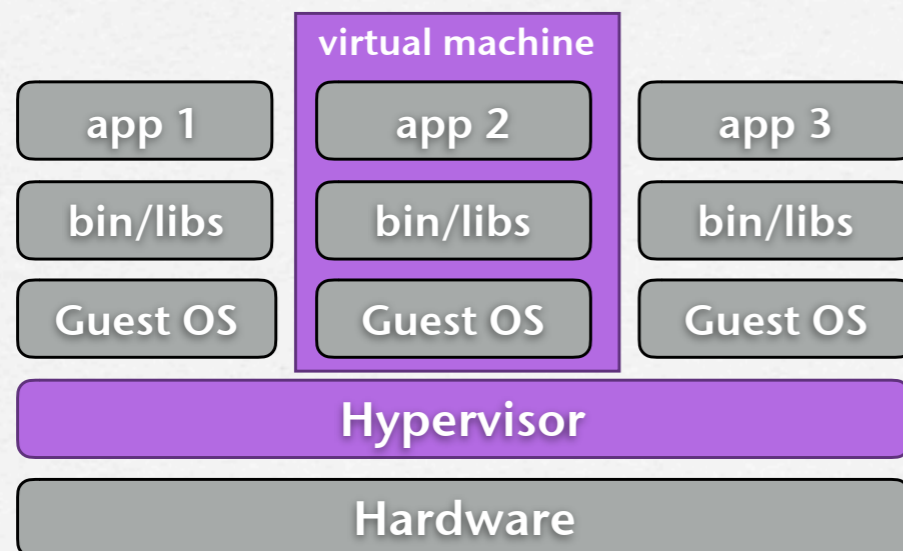
Parallels



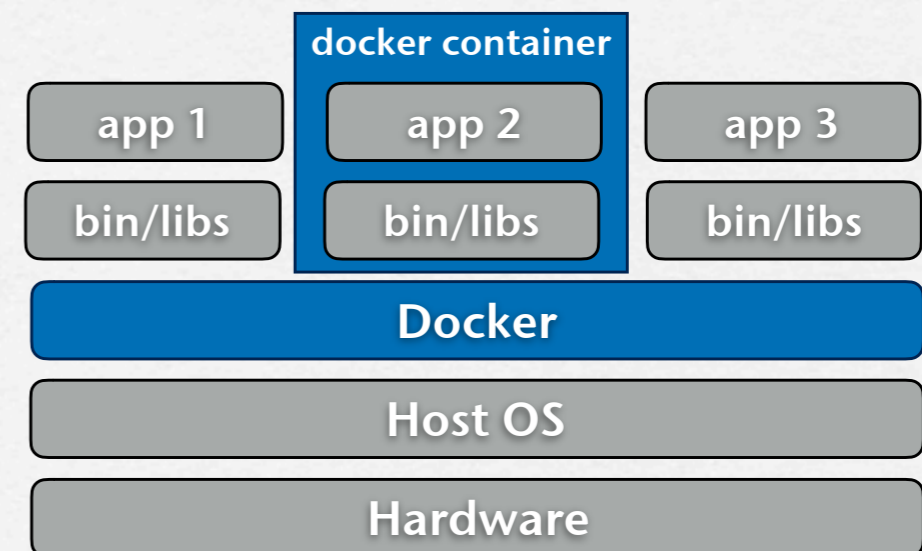
Virtual Machine vs. Containers



The *container-based approach of docker* is a variant of virtualization that tries to be more lightweight and to facilitate / accelerate the deployments of complete systems, i.e., OS + libraries + applications



- large memory footprint (tens of GBs)
- deployment is rather slow
- supports different operating systems on the same hardware



- small memory footprint (tens of MBs)
- deployment is very quick
- supports only one operating system on the same hardware

Cloud Computing | A 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 of Cloud Computing

Peter M. Mell, Timothy Grance

<https://dx.doi.org/10.6028/NIST.SP.800-145>

NIST = National Institute of Standards and Technology, U.S. Department of Commerce

Infrastructure | A Definition as a Service (IaaS)

The capability provided to the consumer is to **provision processing, storage, networks, and other fundamental computing resources** where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications.

The NIST Definition of Cloud Computing

Peter M. Mell, Timothy Grance

<https://dx.doi.org/10.6028/NIST.SP.800-145>

NIST = National Institute of Standards and Technology, U.S. Department of Commerce

Platform as a Service (PaaS) | A Definition

The capability provided to the consumer is to deploy onto the cloud infrastructure **consumer-created or acquired applications** created **using programming languages, libraries, services, and tools** supported by the provider.

The NIST Definition of Cloud Computing

Peter M. Mell, Timothy Grance

<https://dx.doi.org/10.6028/NIST.SP.800-145>

NIST = National Institute of Standards and Technology, U.S. Department of Commerce

Software | A Definition as a Service (SaaS)

The capability provided to the consumer is to use **the provider's applications running on a cloud infrastructure**. The applications are accessible from various client devices through either a **thin client interface**, such as a web browser (e.g., web-based email), or a program interface.

The NIST Definition of Cloud Computing

Peter M. Mell, Timothy Grance

<https://dx.doi.org/10.6028/NIST.SP.800-145>

NIST = National Institute of Standards and Technology, U.S. Department of Commerce

Types of Cloud Services

◆ Infrastructure as a Service (IaaS)

- ◆ Why buy when you can rent and scale?
- ◆ Example: Amazon EC2

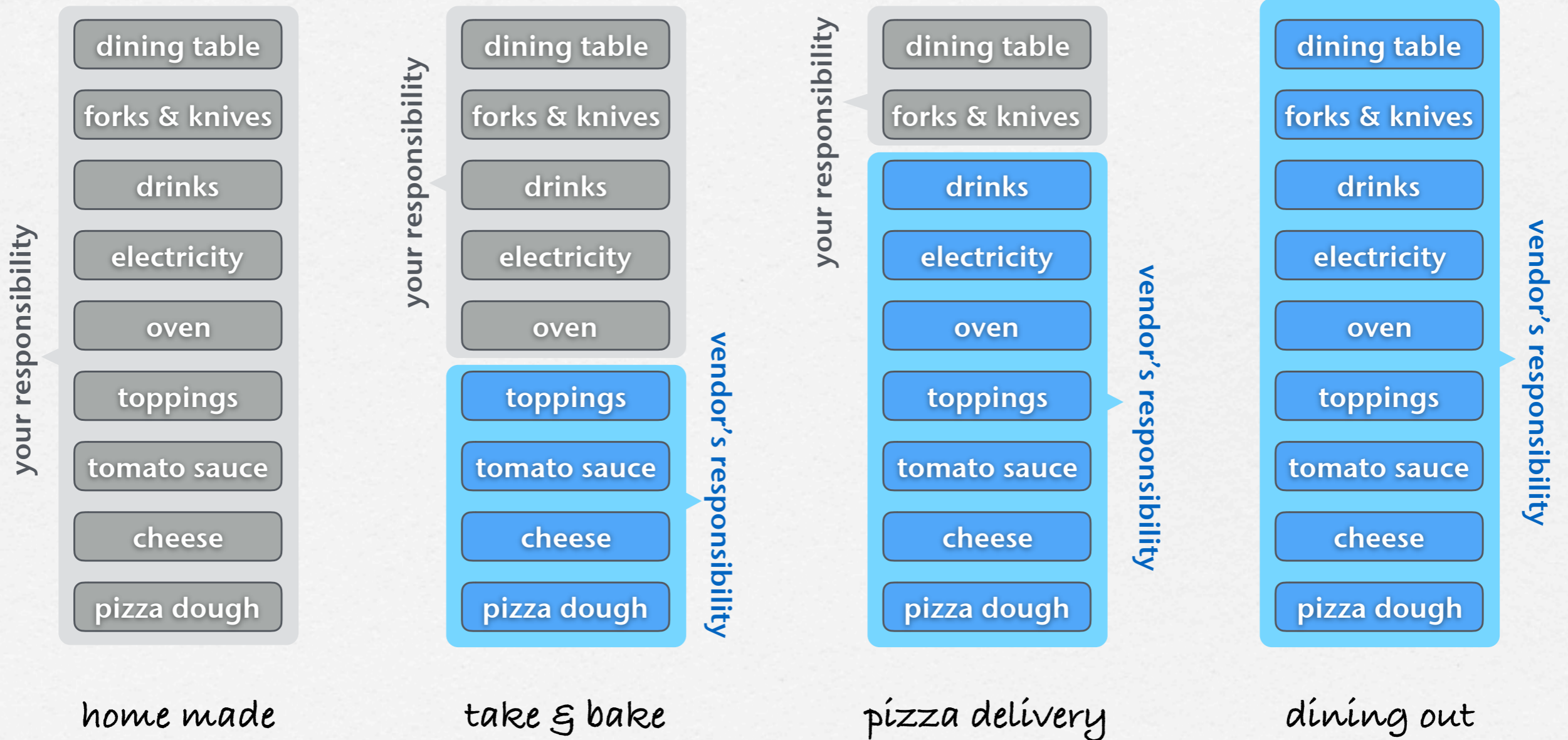
◆ Platform as a Service (PaaS)

- ◆ Give me nice APIs with solid implementations!
- ◆ Example: Matchmore Location-based Pub/Sub

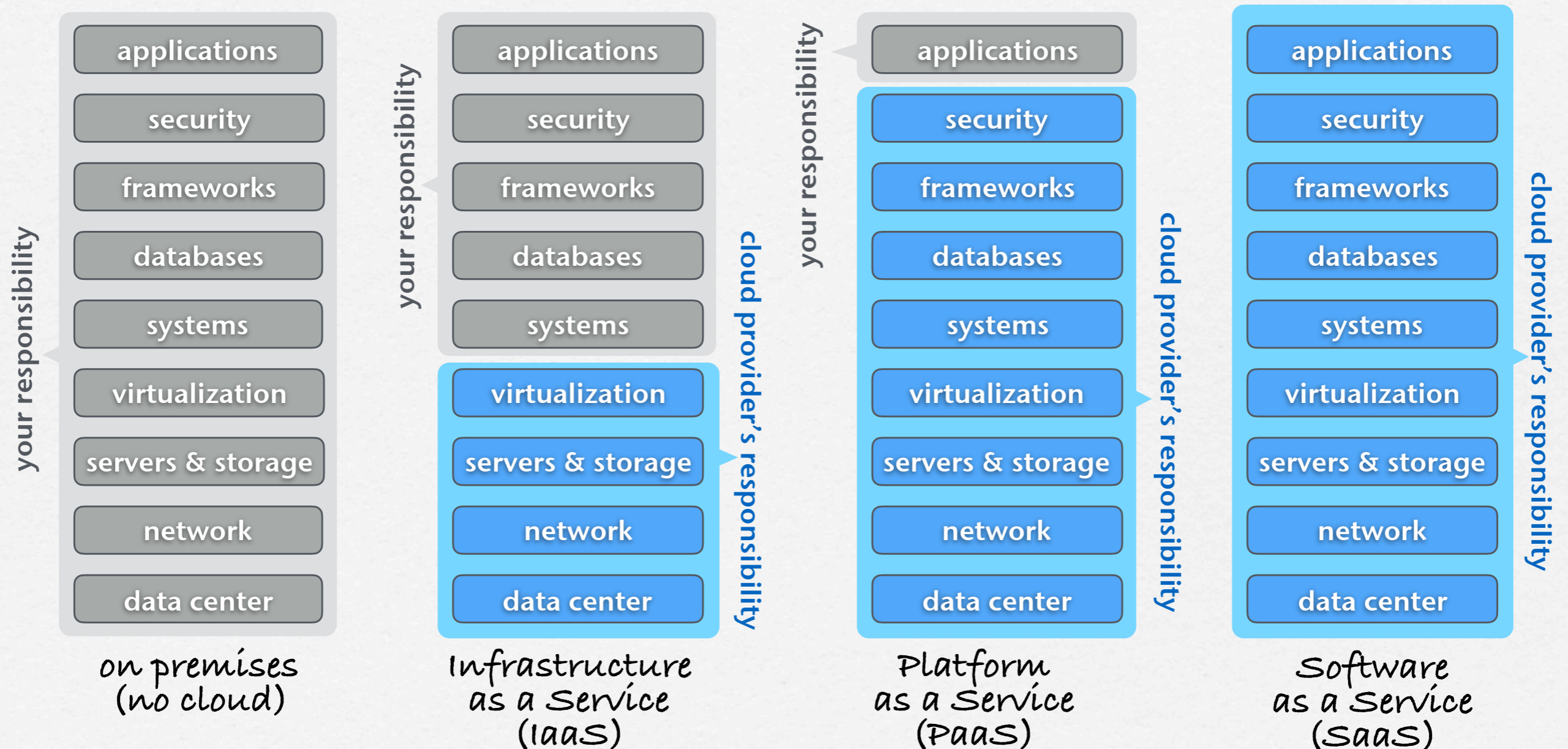
◆ Software as a Service (SaaS)

- ◆ Run it for me and make it accessible anywhere!
- ◆ Example: Google Docs, Sheets, Slides

From pizza...



...to the Cloud !



Cloud Providers



iCloud



Watson



ShareLaTeX



IBM Cloud



GitHub

cloudbees



Eclipse Che



Scaleway

ORACLE[®] Cloud



matchmore ^{BETA}

EXOSCALE



et cæterae...