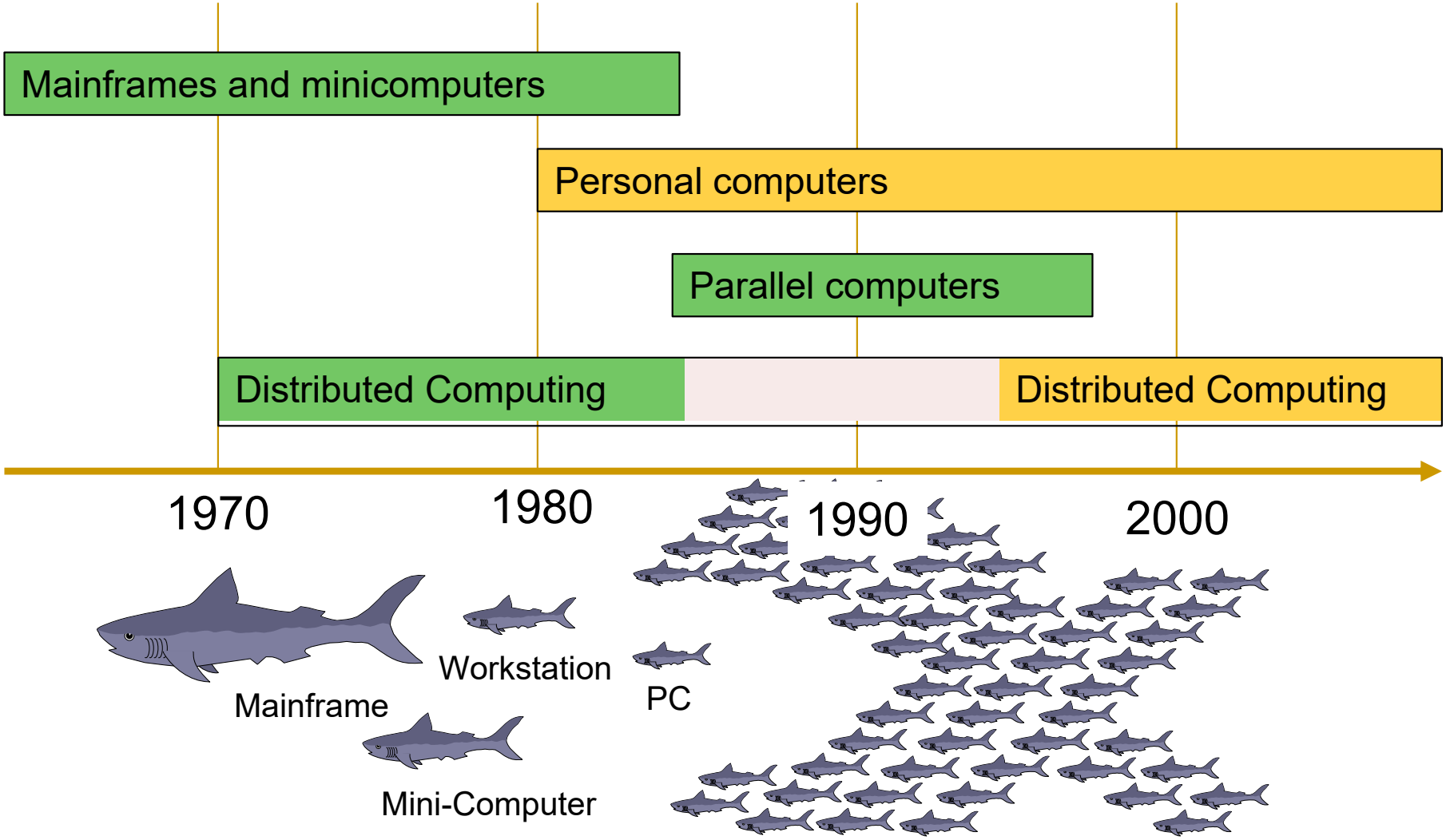
Cloud Computing (intro)

Runtimes for concurrency and distribution

Tullio Vardanega, tullio.vardanega@unipd.it

Academic year 2021/2022

Brief history of computing technologies



Vision of computing utilities

- 1969, Leonard Kleinrock one of the chief scientists of the original ARPANET:

"As of now, computer networks are still in their infancy, but as they grow up and become sophisticated, we will probably see the spread of 'computing utilities' which, like present electric and telephone utilities, will service individual homes and offices across the country."

Referred to as utility computing or, recently (since 2007), as cloud computing:

- users access services based on their requirements without regard to where the services are hosted
- denotes the infrastructure as a "cloud" from which businesses and users can access applications as services from anywhere in the world and on demand
- cloud computing can be classified as a new paradigm for the dynamic provisioning of computing services supported by state-of-the-art data centers employing virtualization technologies for consolidation and effective utilization of resources.

What is Cloud Computing /1

- Diverse kinds of computing needs
  - Scientific (high-performance) computing
    - Massive, application-specific, often brute-force
    - Compute-power hungry
  - Data centers
    - Focus on large-scale data processing, performing data-parallel computations on large volumes of data objects,
      - Mining billions of web pages, classifying data …
  - Web-based computing and service computing
    - Turning the Internet into a rich-application and service-delivery platform

What is Cloud Computing /2

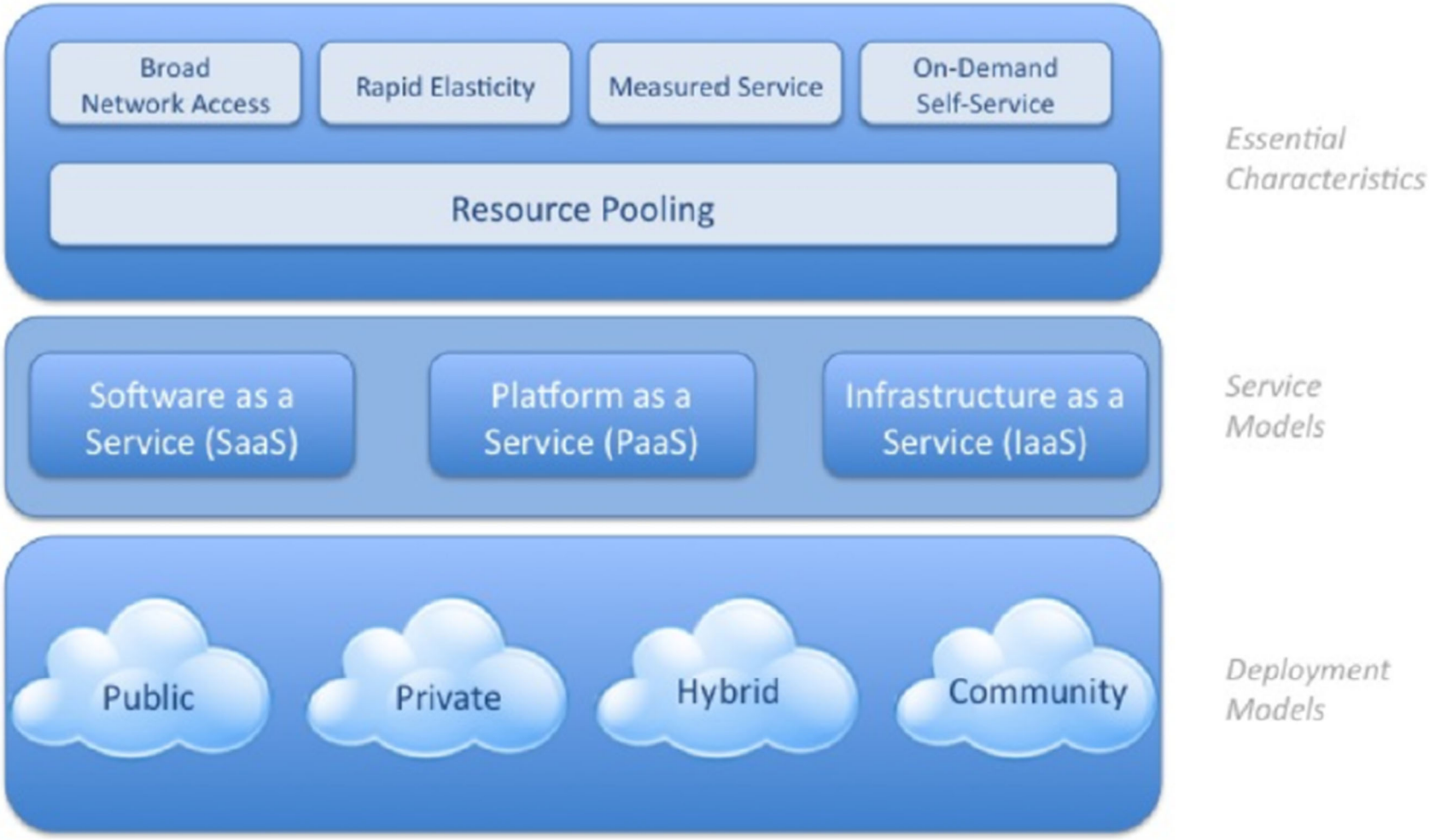
- The Guardian, Sept. 29, 2008
  - Richard Stallman, Founder, Free Software Foundation
  - "*It's worse than stupidity: it's marketing hype. Somebody is saying this is inevitable - and whenever you hear that, it's very likely to be a set of businesses campaigning to make it true*."
- Wall Street Journal, Sept. 26, 2008
  - Larry Ellison, CEO, Oracle
  - "*...we've redefined Cloud Computing to include everything that we already do... I don't understand what we would do differently ... other than change the wording of some of our ads*."

What is Cloud Computing /3

- National Institute of Standards and Technology, US
  - Cloud computing is a model for enabling **[1]** convenient **on-demand**, **[2]** **pay-per-use**, **[3]** **networked access** to a *shared pool of configurable computing resources* (e.g., networks, servers, storage, apps, and services) that can be **[4]** **rapidly provisioned** and released with **[5]** minimal management effort or service provider interaction (**self-service**)
  - This model promotes availability and is composed of the five essential characteristics listed above, three service models (**SaaS**, **PaaS**, **Iaas**), and four deployment models (private, community, public, hybrid)

http://www.nist.gov/itl/cloud/upload/cloud-def-v15.pdf

What is Cloud Computing /4

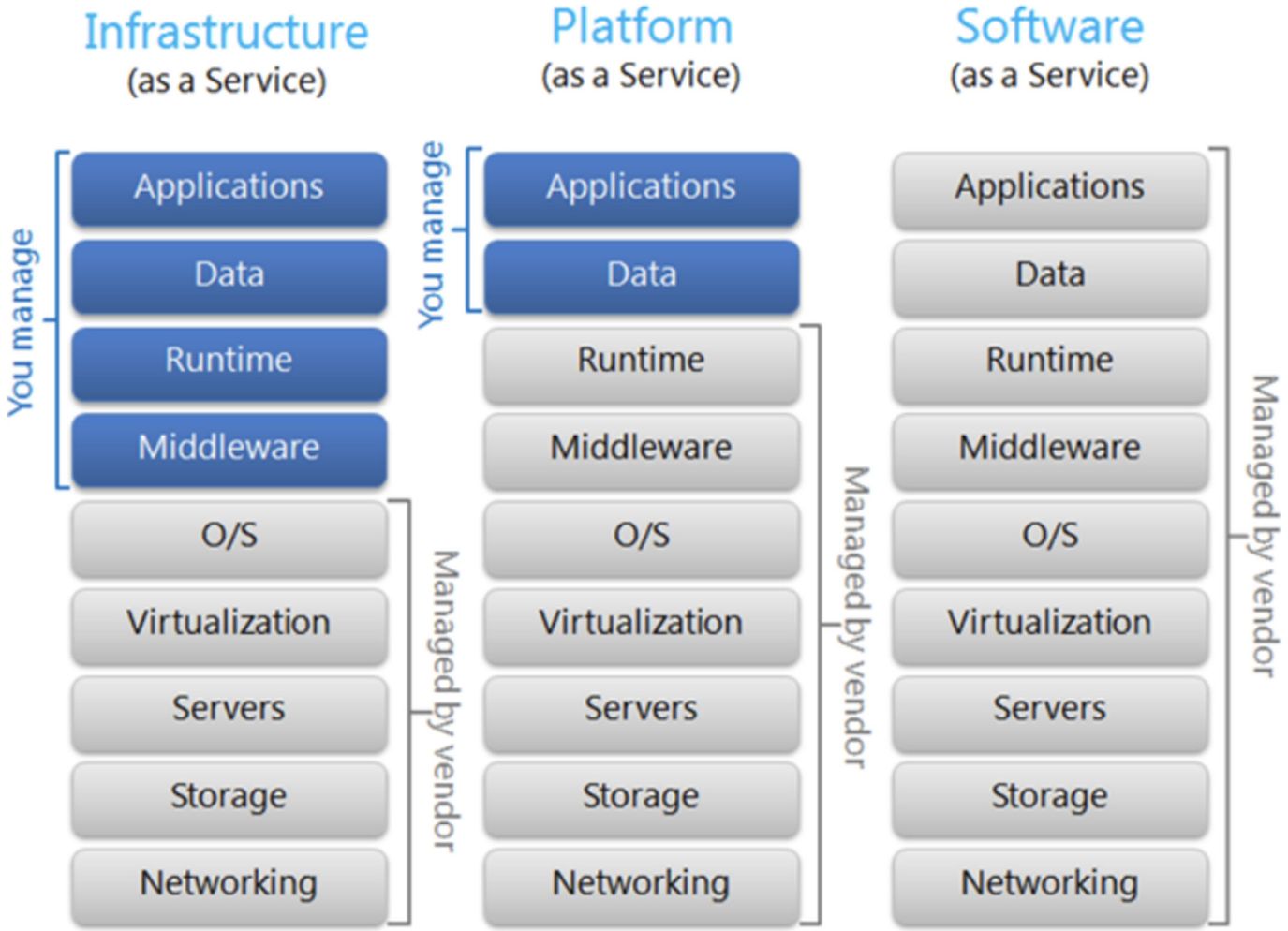


What is Cloud Computing /5

- Over-the-Internet provisioning of dynamically scalable, virtualized resources
  - Computing, storage and network resources; computing platform and middleware; services (**IaaS**, **PaaS**, **SaaS**)
- Users do not need expertise in (computing) resource management
  - Hardware maintenance, system configurations, software upgrades, information updates, etc.
- Pay-per-use, like other utilities

"I don't care where my servers are, who manages them, where my documents are stored, or where my applications are hosted. I just want them always available and access them from any device connected through Internet. And I am willing to pay for this service for as a long as I need it."

Cloud Computing service models /1

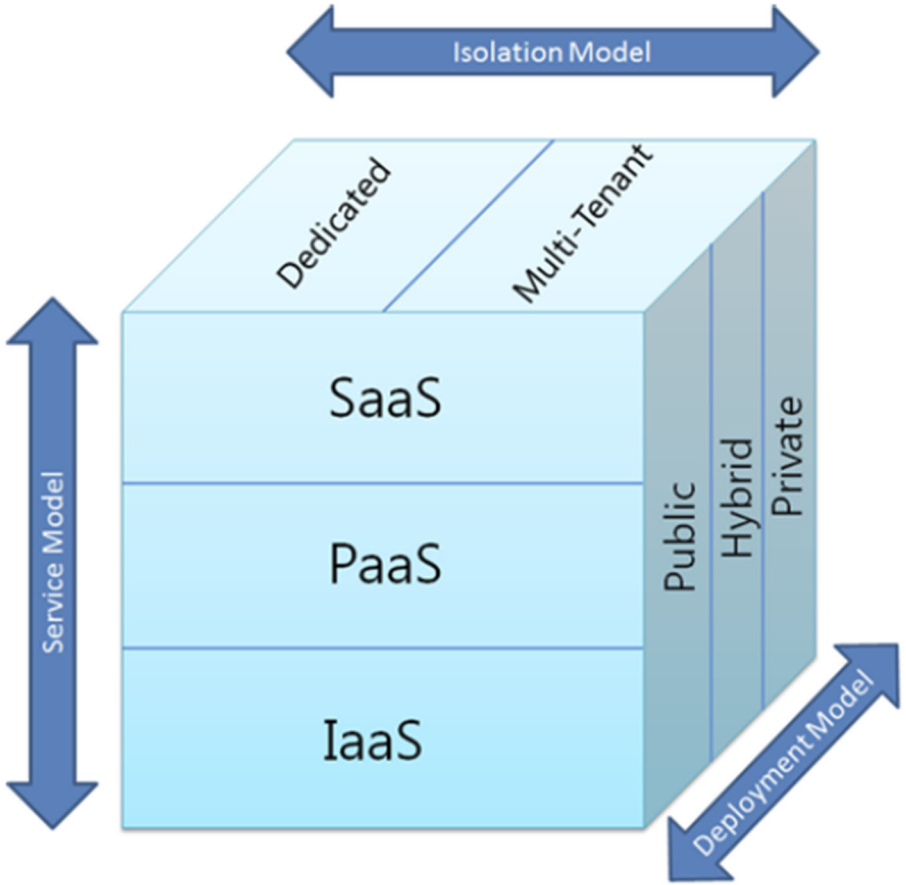


Fonte: http://goo.gl/1jmkR

Cloud Computing service models /2

- **Infrastructure as a service**
  - *Computing resources*, such as storage and processing
    - You have your own program, but do not have the needed computing facility ⇒ use Amazon EC2
    - You have lots of data but insufficient local storage ⇒ use Amazon S3
- **Platform as a service**
  - *Support layer* to help users focus on their software development or management tasks (w/o the hassles of resource provisioning)
    - Ranges from handling applications (as in an application server) to developing and deploying them
- **Software as a service**
  - *Application delivery* model where software and its associated data are hosted centrally and are accessed remotely via clients running on web browsers over the Internet

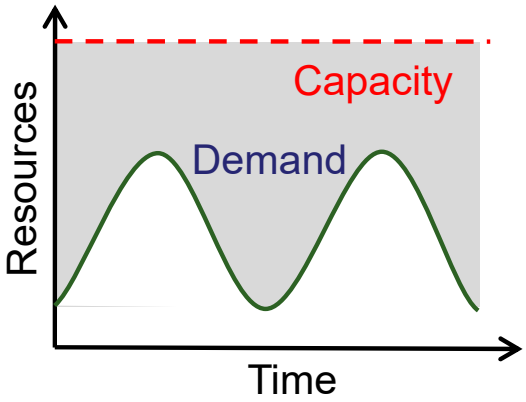
Dimensions of Cloud Computing



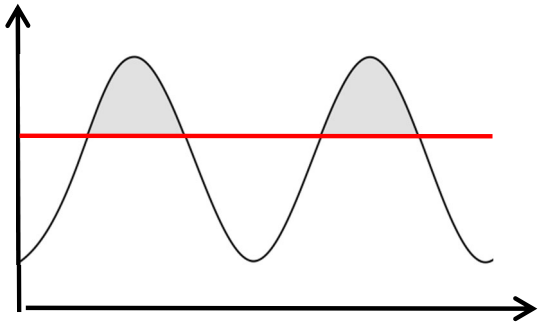
Fonte: http://goo.gl/1jmkR

IaaS requirements /1

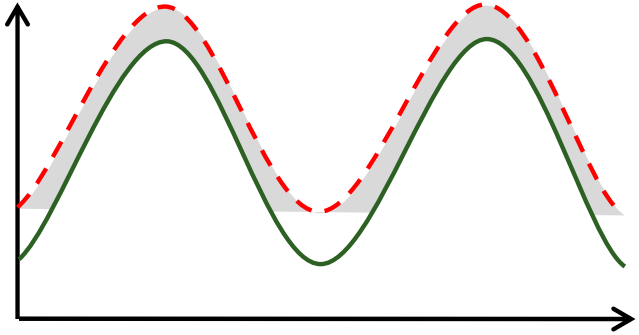
- **Scalability**
  - System performance should stay roughly the same against small-scale or large-scale demand
- **Elasticity**
  - Resource provisioning should be done only for as long as needed and strictly to the extent required



Over-provisioning



Under-provisioning



Elastic scalability

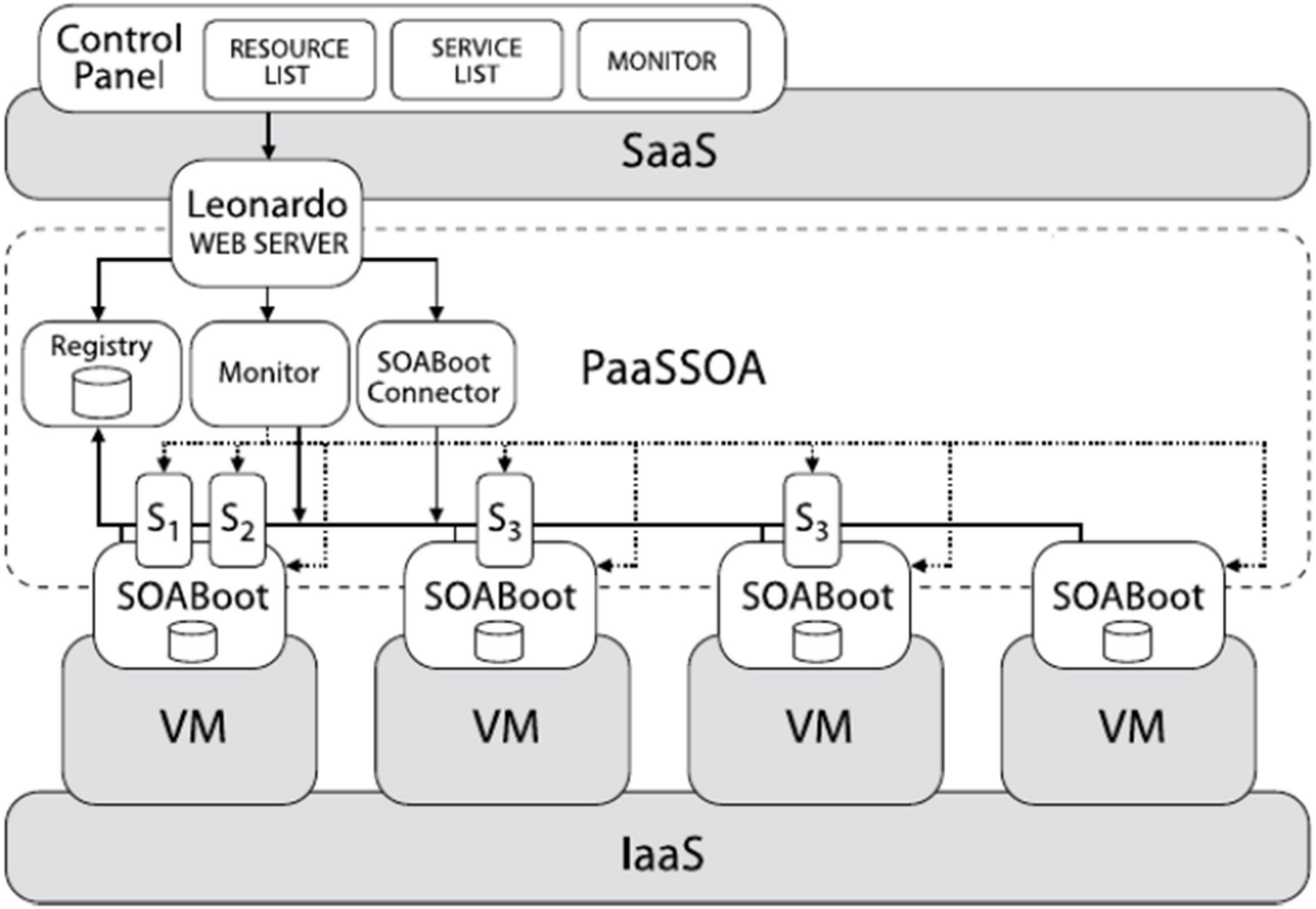
IaaS requirements /2

- **Availability & reliability**
  - Clients should not worry about provider-side failures
    - All such failures should be masked
  - Data stored anywhere in the Cloud can be retrieved whenever needed
  - Communication capability and capacity within the provider domain should be always available

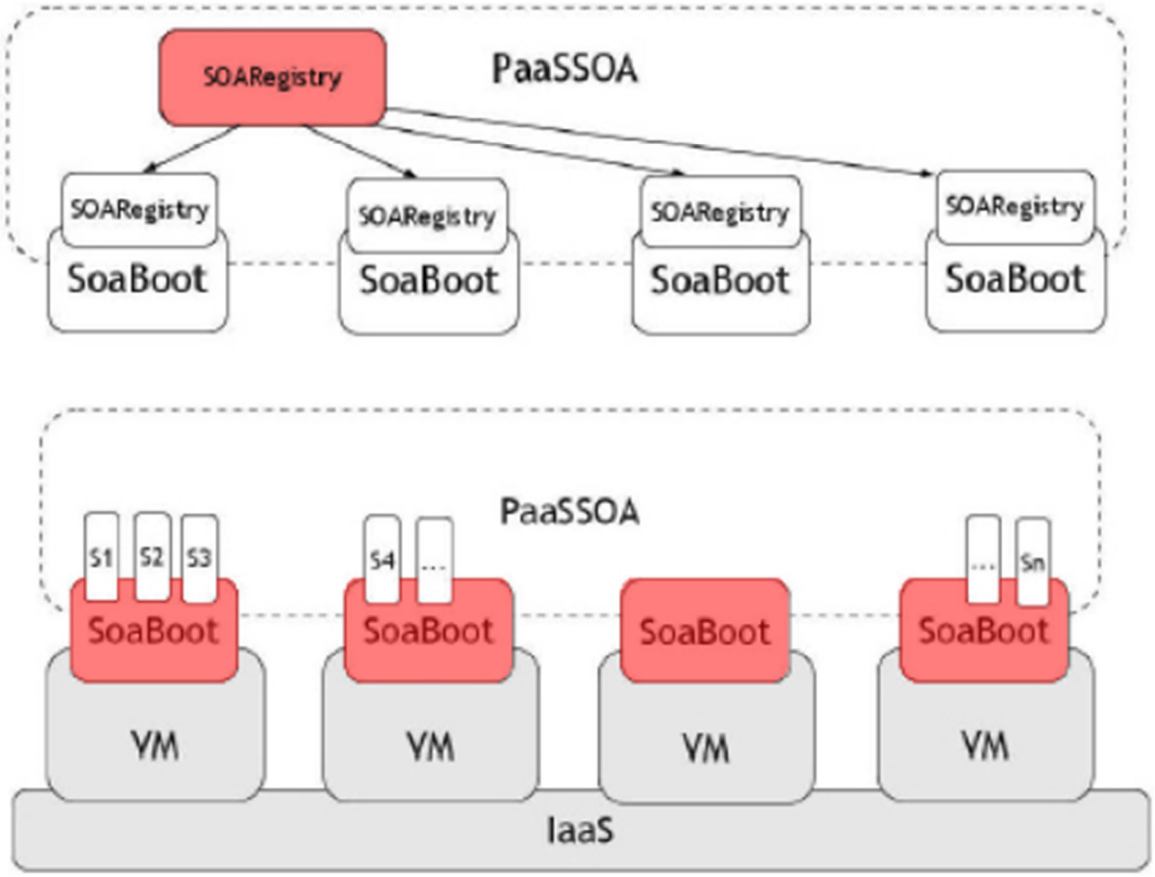
IaaS enabling technologies

- **Virtualization**
  - Cloud does not require virtualization, but it was technically enabled by it
  - Virtualization achieves efficient and secure sharing
    - VMs provide natural isolation
    - VMs can be easily handled, deployed, migrated and assigned resources
- **Storage**
  - File systems for a large number of users
    - GlusterFS (https://www.gluster.org/), Amazon S3
  - Structured data storage in peta-scale
    - Big table, Distributed Hash Table solutions
- **Monitoring, debugging, dynamic adaptation**
  - Performance analysis, fault isolation
  - Event notification
  - SLA negotiation and assurance

Inside view of a PaaS /1

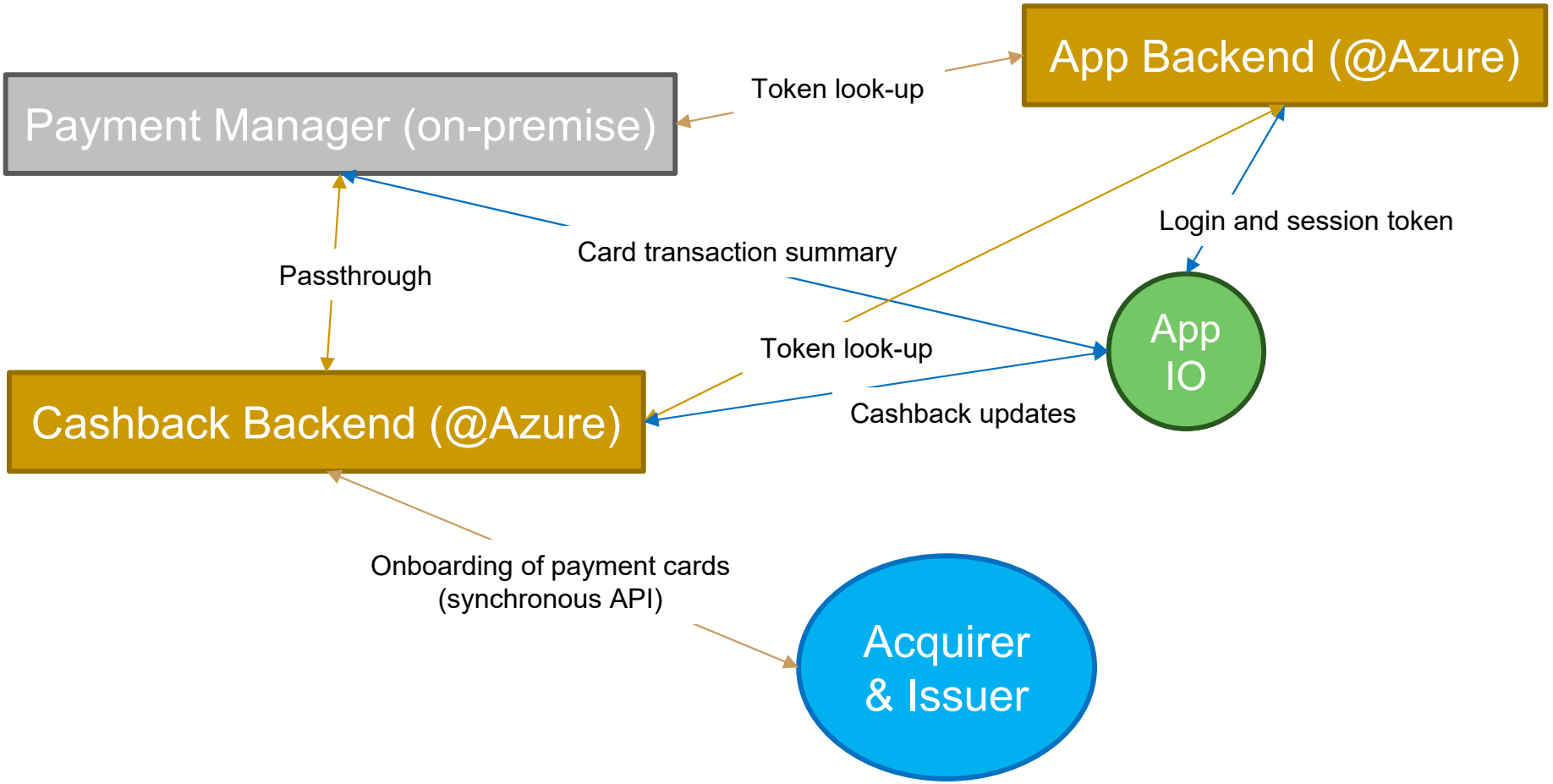


Inside view of a PaaS /2



- PaaS handles a federated pool of computing resources
  - 1st-level registry dispatches incoming requests
  - And resolves names to 2nd-level
- SOABoot: service container assigned to VM hosts
  - Individual services can be queried, deployed, retired, started, stopped

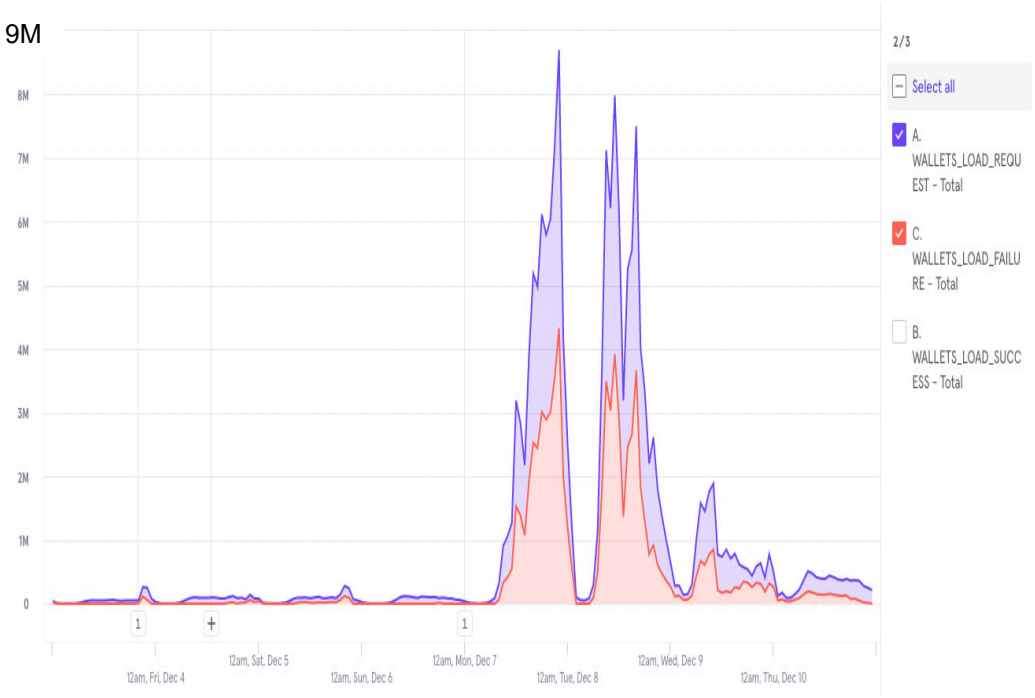
The national *cashback* use case /1



https://medium.com/pagopa-spa/cashback-retrospettiva-su-un-avvio-sfidante-con-lo-sguardo-rivolto-al-futuro-206cb609e4bb

The national *cashback* use case /2

- December 7, 2020 end of day: **6.6M** downloads of App IO
  - In the next 24 hours, **+1M**
- **14K** operations/second on the App IO, to activate service and upload card data
  - **2.3M** active users
- **2.7K** ops/second on the Backend



Peak of requests

The national *cashback* use case /3

- Massive overload of Payment Manager
  - Mostly caused by lock-based access to secure data
- Massive overload of 3D-Secure verification
  - Volume of traffic 3x higher than normal daily average
- Baseline scenario for stress testing referred to Black Friday case
  - Actual volume was 2x higher
- Harshest problem caused by user and app behaviour on access or transaction failure
  - Well-known "*retry management problem*" triggered by transient faults

Cloud-fit retry patterns

- **Exponential back-off**: variable delays between retries levels peak load
  - https://dzone.com/articles/understanding-retry-pattern-with-exponential-back
  - https://aws.amazon.com/blogs/architecture/exponential-backoff-and-jitter/
- **Circuit breaker**: for transient faults that may last long, return immediate failure to stop requests from accumulating
  - https://martinfowler.com/bliki/CircuitBreaker.html
- **Backpressure**: giving input queues a maximum size stops excess requests from being accepted (and the client sees there is a problem)
  - https://www.tedinski.com/2019/03/05/backpressure.html
- **Throttling**: rejecting selected requests to lower pressure on system
  - https://docs.microsoft.com/en-us/azure/architecture/patterns/throttling
  - https://aws.amazon.com/premiumsupport/knowledge-center/dynamodb-table-throttled/