

# I datacenter e la migrazione al cloud

Docente:

Antonio Servetti (DAUIN - Politecnico di Torino)

email: [antonio.servetti@polito.it](mailto:antonio.servetti@polito.it)

Materiale [online](#) su GitHub

# Sommario

- Introduzione
  - definizione, obiettivi
  - vantaggi e svantaggi
- Datacenter
  - tipologie, zone e regioni
  - provider cloud qualificati
- Architettura software
  - stack applicativo, virtualizzazione, microservizi
  - modelli di servizio
- Migrazione
  - roadmap, analisi costi-benefici
  - strategie di migrazione

# Strumenti

- Strategia Cloud Italia
- Manuale di Abilitazione al Cloud
- Il glossario del cloud



# Introduzione

---

Il *Codice dell'Amministrazione Digitale* (CAD), individua il digitale come modalità principale per assicurare *disponibilità, gestione, accesso, conservazione* di dati, documenti e informazioni.

---

# Motivazioni

*L'obbligo di migrazione al cloud per le PA con l'articolo 33 septies del decreto legge n.179 del 2012 con il consolidamento e la razionalizzazione dei siti e delle infrastrutture digitali del Paese.*

Il principio *cloud first* mantenuto nell'aggiornamento del Piano Triennale al 2024 che stabilisce che *in fase di definizione di un nuovo progetto e di nuovi servizi le PA adottino il cloud come "paradigma di riferimento"*.

La [Strategia Cloud Italia](#) che indica i passi amministrativi necessari all'implementazione di soluzioni cloud con attenzione a *tutela della privacy, autonomia tecnologica, controllo sui dati e resilienza del sistema*.

Il [Manuale di Abilitazione al Cloud](#) che accompagna le PA nell'identificazione degli applicativi da migrare fino alla valitazione della migrazione stessa.

# Obiettivi

**Availability (disponibilità).** Il sistema deve essere sempre online sia che manchi la luce, sia che si danneggi un dispositivo di rete, sia che si rompa un disco.

**Disaster recovery.** Il sistema deve "riprendersi" nel minor tempo possibile (meglio se immediatamente) da danni strutturali come incendi o allagamenti.

**Scalabilità.** Capacità di gestire carichi di operatività adattando le risorse in base al bisogno (orizzontale: più macchine, verticale: più risorse).

**Backup.** Ci deve essere la certezza di non perdere dati, quindi di poter riprendere l'attività, in caso di corruzione dei dati o hacking (ransomware)

Altro: resilienza (monitoraggio, bilanciamento, design for failure), sicurezza, privacy, alerting, automazione

# Definizione

[Cos'è il cloud - AGID](#). Il cloud computing (in italiano nuvola informatica), è un modello di infrastrutture informatiche che consente di disporre, tramite internet, di un insieme di risorse di calcolo (ad es. reti, server, risorse di archiviazione, applicazioni software) che possono essere rapidamente erogate come servizio.

**Analogia server / bestiame.** I PC e in generale le risorse cloud sono gestite come bestiame, al contrario delle proprie che si tendono a considerare come animali domestici (ci si affeziona).

[1] ["Cloud Computing"](#), NIST, 2011

[2] [Pets vs Cattle Analogy](#), 2016

## Caratteristiche (alcune)

**Accesso attraverso la rete.** Si assume che la rete (Internet o intranet) sia disponibile e a banda larga.

**Self-service, on-demand.** Le risorse sono a disposizione del cliente secondo necessità, senza ritardi, quando servono e se gli servono (flessibilità).

**Elasticità.** Il servizio può scalare rapidamente a seconda delle necessità del cliente.

**Pagamento a consumo.** Il cliente paga solo ciò che usa evitando investimenti iniziali.

**Delega.** La gestione e la conoscenza delle risorse, a vari livelli, è demandata al Cloud Provider.

ANALOGIA  
AUTONOLEGGIO



## Vantaggi

**Riduzione dei costi (a breve termine)** Da un punto di vista tecnico ... ottenere gli stessi risultati di scalabilità, disponibilità e tolleranza agli errori e sicurezza in proprio ha un costo "spropositato": investimenti, tempo, personale, aggiornamenti continui, sicurezza. Attenzione però al TCO.

**Flessibilità e scalabilità** Acquistare risorse in proprio ha anche *tempi* più elevati, richiede di anticipare i costi, ha dei limiti fisiologici, non si adatta a "carichi" o "crescite" variabili. ... opportunità di migliorare la qualità dei propri servizi.

**Democratizzazione delle risorse** Accesso alle risorse anche per piccole realtà (es. equivalenza tra 1 macchina per 1000 ore o 1000 macchine per 1 ora)

**Delega** Nella gestione degli aggiornamenti, del supporto utente, delle attività manuali a basso valore (installatori), adeguamento normativo di sicurezza e privacy.

[1] ["Perchè usare il cloud"](#), MAC

## Svantaggi



La modernità delle piattaforme cloud non deve far pensare che non esistano svantaggi, come:

- dipendenza dalla rete (Internet)
- perdita del controllo sui dati
- lock-in (dipendenza dal fornitore)
- delega della gestione, ma responsabilità del servizio

## Dipendenza da Internet

L'uso del cloud implica necessariamente l'utilizzo della "rete" (interna o pubblica) che quindi deve essere *sempre disponibile, a basso ritardo e a banda larga.*

Insorgono problemi in caso di

- zone poco "coperte",
- connessioni non ridondanti,
- applicazioni critiche (controllo di sistemi fisici come l'apertura di porte o accensione di luci),
- applicazioni che generano moli enormi di dati (e.g. editing video).



## Perdita del controllo sui dati

Rischio di *loss of control* cioè della perdita del controllo sui dati messi in cloud, anche in funzione della responsabilità riconosciuta dalle normative (v. GDPR) non solo in termini di integrità, disponibilità e resilienza, ma anche di **riservatezza** (e protezione dati personali).

- Verifica della compliance alla normativa privacy da parte del fornitore del servizio (Service Legal Agreement)

## Lock-in

Fenomeno di natura tecnica ed economica tale per cui una organizzazione non riesce a svincolarsi facilmente da una scelta tecnologica precedentemente effettuata.

**Lock-in sul dato.** Impossibilità di esportare i dati in modo automatico e in formati *standard* per essere importati in altri sistemi.

**Lock-in tecnico.** Certe funzionalità non sono disponibili (o lo sono in modo diverso) su tutti i provider cloud.

**Lock-in sulla conoscenza.** Il know-how nell'uso di una piattaforma cloud non necessariamente è trasferibile (riusabile) per un'altra piattaforma.



# Mitigazione del lock-in

## Pratiche tecnologiche

- accesso ai servizi tramite API e protocolli standard
- formati di dati standard e aperti con la possibilità di esportare i dati
- utilizzare framework di sviluppo e stack tecnologici open
- evitare l'utilizzo di servizi "unici" del cloud service provider
- proprietà e accesso al codice sorgente

**Pratiche metodologiche.** Orientate alla condivisione della conoscenza: *documentazione* (traccia delle decisioni significative architetturali e di business, architettura software, procedure, guide, ecc.) e *metodologia di lavoro* che segua buone pratiche di collaborazione e condivisione delle informazioni. [MAC]

**Pratiche contrattuali.** Introdurre nei documenti di gara le clausole relative alle buone pratiche di cui sopra.

# Responsabilità

Attenzione ai contratti (Service Level Agreement), sono previste penali o no per

- interruzione dei servizi
- accesso non autorizzato di terzi ai dati
- corruzione o perdita di dati, o errori nei backup
- cessazione o dismissione del servizio, o di un software o di una versione dello stesso



# Datacenters

Struttura fisica che ospita l'infrastruttura cloud: server, dischi, reti, condizionamento, alimentazione.



- [1] "[Data Center design: dal mainframe al cloud](#)", PSN, 2/2/2024
- [2] "[Uno sguardo all'interno dei data center di Google](#), maps / streetview, 2012

## Tipi di datacenter

Detti anche **modelli di deploy** i tipi di datacenter possono essere differenziati come segue:

- Privato (locale) / on-premise
- Colocation / multi-tenant
- Pubblico
- Altro: gestito / managed, community, multi-cloud, ibrido



# Datacenter privato (locale), on-premise

Cosa

- l'infrastruttura è di proprietà del cliente stesso, non necessariamente "in sede", comunque connessa tramite Internet

Perchè

- elevate disponibilità (pago tutto subito, si deprezza nel tempo)
- esigenza di controllo (anche fisico) e sicurezza delle informazioni
- esempio: polo strategico nazionale

Responsabilità

- dei locali, dell'elettricità, del condizionamento, dell'acquisto e gestione dell'hardware, della manutenzione e monitoraggio del software

[1] ["Cloud privato"](#), Manuale Abilitazione Cloud

# Datacenter in colocation (housing)

Cosa

- si affitta uno spazio server presso terzi, ma l'accesso è esclusivo.

Perchè

- esigenza di controllo e sicurezza
- delegare a terzi la gestione di corrente ed emergenze (disaster recovery fisico)

Responsabilità

- tutto, ma solo su server (hw) e software

# Datacenter pubblico

Cosa

- si demanda la gestione (almeno fino al livello hardware) a terzi (detti hyperscaler: Amazon, Microsoft, Google)
- differenti livelli di gestione: infrastruttura, piattaforma, software

Perchè

- non vincoli stringenti di sicurezza (delegabili a terzi)
- scalabilità on-demand
- contratti self-service, no-delay (basta un click per usare il doppio dei server)

[1] "[Cloud pubblico](#)", Manuale Abilitazione Cloud

## Datacenter "altre tipologie"

**Gestito (managed)** Il cliente noleggia l'hardware che però è distribuito, supervisionato e gestito da un fornitore di servizi terzo.

**Community** L'infrastruttura è condivisa tra più organizzazioni "gemellate" che si dividono i costi.

**Multi Cloud** L'ambiente è distribuito in modo eterogeneo su più cloud pubblici, secondo i fabbisogni senza "legami".

**Ibrido** Una combinazione di privato e pubblico in cui certi dati/servizi sono mantenuti "in house" per maggiore sicurezza.

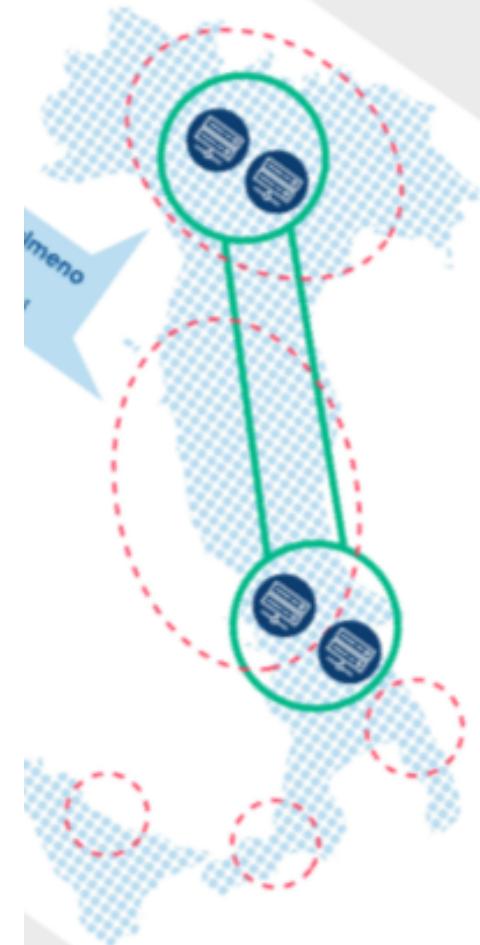
# Zone e Regioni

## Zona ("single failure domain")

- datacenter singolo con propria alimentazione, rete, condizionamento
- che è *indipendente* dagli altri
- eventualmente su più edifici

## Regione insieme di zone, tipicamente 3, "in collaborazione"

- in prossimità geografica (tra 10-100 km) tra di loro
- collegate da reti ad elevatissima velocità, bassa latenza e ridondanti per la replica dei dati in tempo reale



# Zone e Regioni (availability)

Le applicazioni devono essere distribuite su *più* zone all'interno di una regione per assicurare "high availability" (elevata disponibilità)

- se una zona "va giù" si è rediretti o si attiva la replica in una altra zona

In caso di catastrofi naturali o disservizi significativi può cadere anche una intera regione, occorre valutare anche repliche (di "recupero" più che di "continuità") in altre regioni.

- le regioni sono invece ad elevata distanza tra di loro



# Polo strategico nazionale

Infrastruttura informatica a servizio della PA localizzata sul territorio nazionale (autonomia) e distribuito geograficamente.

- sedi di Acilia e Pomezia nel Lazio, Rozzano e Santo Stefano Ticino in Lombardia
- 4 data center allestiti in configurazione di *doppia region\** (nord - sud) in *dual availability zone*
- *continuità di servizio infrastrutturale* tra i due data center della stessa regione
- *reagire ad eventi catastrofici tramite la riattivazione dei workload* su una AZ diversa

PSN attivo dal 2023, la gestione affidata a un fornitore qualificato partecipato da TIM, Leonardo, Cassa Depositi e Prestiti, e Sogei.

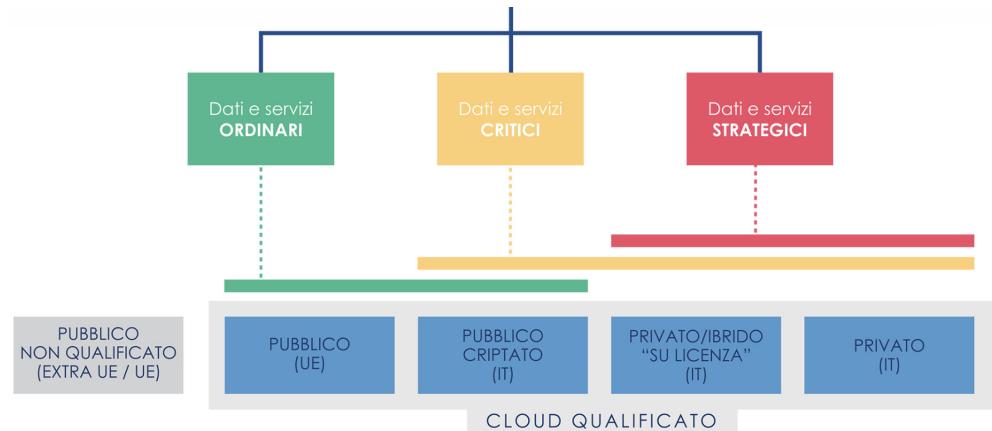
Obiettivo: migrazione dati e servizi **critici delle amministrazioni centrali, ASL** (sembra i comuni non rientrino in questa categoria)

[1] <https://www.polostrategiconazionale.it/>

# Servizi cloud qualificati

Individuazione ex-ante di provider che soddisfino i requisiti ACN (Agenzia per la Cybersicurezza Nazionale)

- cloud pubblico con dati localizzati UE compatibile con legislazione (GDPR, NIS) e rispetto requisiti di sicurezza tecnico-organizzativi
- cloud pubblico con dati localizzati IT e controllo della cifratura in Italia
- cloud privato e ibrido di fornitore soggetto a vigilanza pubblica
  - tecnologia "su licenza" da un hyperscaler
  - tecnologie commerciali scrutinate (privato)



Non qualificati sono quelli pubblici extra-UE.

# Classificazione dei dati

In base all'effetto della compromissione di servizi e dati digitali:

**Strategico.** Può avere un impatto sulla sicurezza nazionale

**Critico.** Può causare un rischio al mantenimento di funzioni rilevanti per tutti gli aspetti della vita del Paese (salute, sicurezza pubblica, benessere economico e sociale): assistenza sanitaria di base, pronto soccorso, day surgery, numeri unici

**Ordinario:** Non genera l'interruzione di servizi dello Stato o, comunque, non pregiudica il benessere economico e sociale del Paese: assistenza termale, veterinaria, medico legale

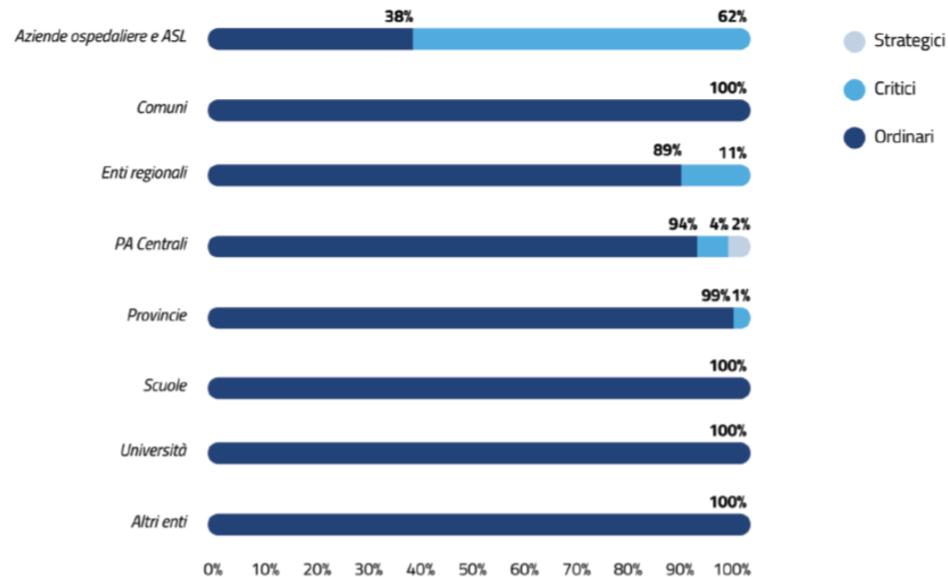
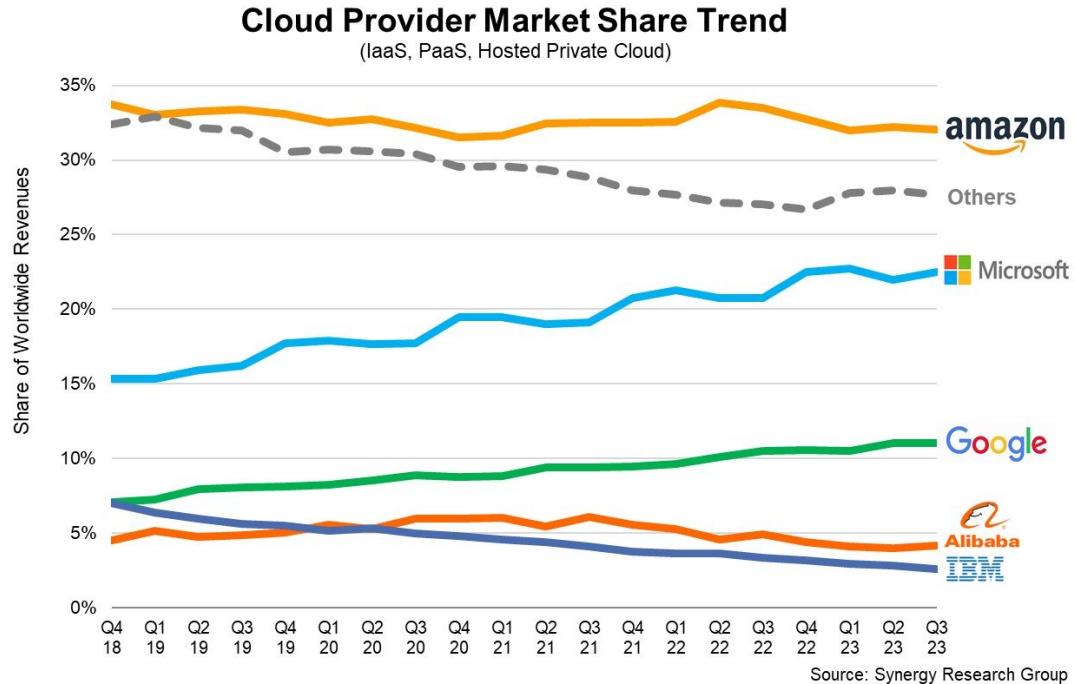


FIGURA 41 – PERCENTUALE DI SERVIZI CLASSIFICATI COME ORDINARI, CRITICI E STRATEGICI, PER CATEGORIE DI PA

# Cloud marketplace

All'interno del "Mercato elettronico dei servizi cloud qualificati" si trovano

- grosse società italiane (TIM),
- "in-house" regionali (CSI Piemonte),
- hyperscaler internazionali (AWS, Azure, Google, ecc.)

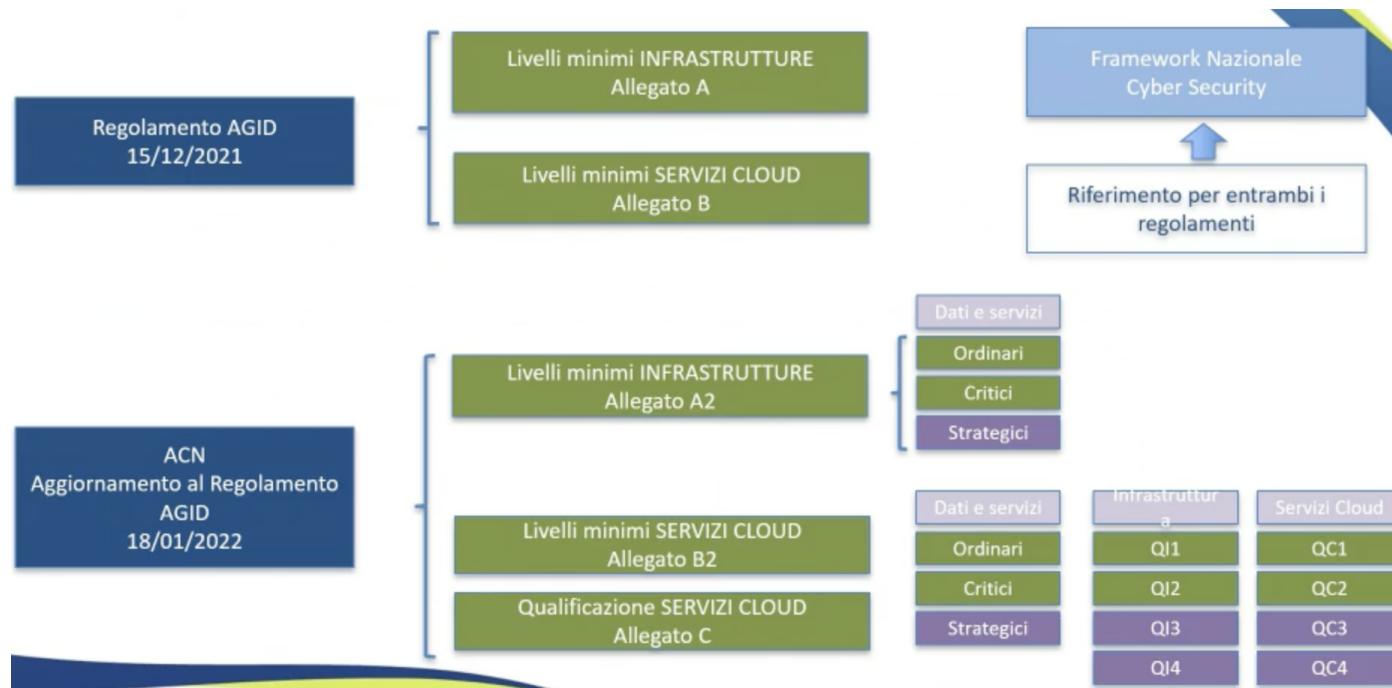


## [1] "Cloud Marketplace"

Il marketplace comprende anche servizi di consulenza per l'assessment e la migrazione.

# Requisiti AGID e ACN

Strategia Cloud Italia per identificare i data center in possesso dei requisiti minimi per erogare servizi e classificarli in base alla tipologia di servizi che possono gestire (ordinari, critici, strategici)





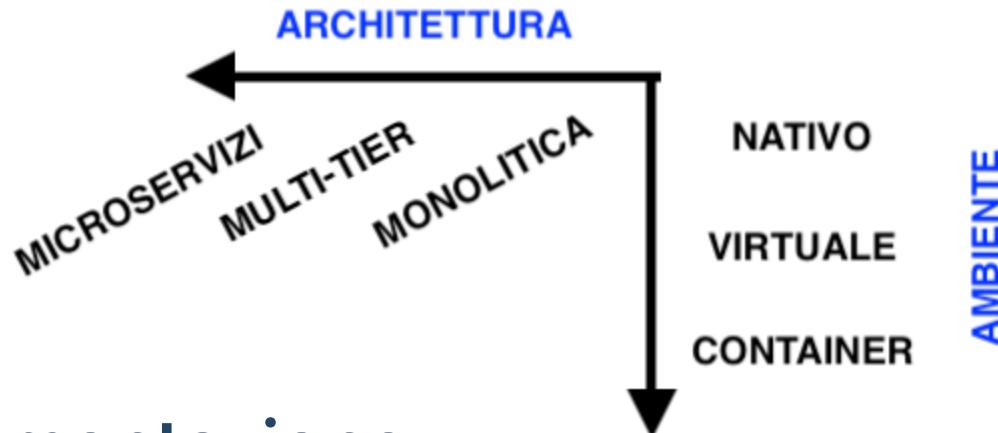
## Architettura software

---

Al "di sopra" del livello *infrastrutturale* fisico abbiamo lo stack, cioè i livelli, software degli applicativi cloud la cui gestione può essere progressivamente demandata al provider stesso.

### Livelli:

rete, archiviazione, cpu, sistema operativo, middleware, runtime, dati, applicazioni



## Virtualizzazione e frammentazione

L'evoluzione degli applicativi software ha favorito due tendenze la *virtualizzazione* degli ambienti di esecuzione e la *frammentazione* dei compiti.

La virtualizzazione permette di "scollegare" l'ambiente di esecuzione dall'hardware e in generale dal "contesto" che sottende e circonda l'applicazione.

La frammentazione permette di separare i compiti dell'applicazione in unità software distinte di più facile sviluppo, gestione e manutenzione.

Applications

Data

Runtime

Middleware

O/S

## Livello cosiddetto "hardware"

La parte software dei sistemi informatici "gira" su:

- server: la macchina fisica, più che altro intesa come capacità di calcolo (CPU) e memoria (RAM)
- storage: i dischi per l'archiviazione dei dati
- networking: l'infrastruttura di rete per comunicare con gli altri servizi e con l'utente

Se mettiamo tutto insieme abbiamo lo schema a lato.

# Stack di sistema

L'applicazione stessa "appoggia" su componenti esterne:

- i dati o "base dati": generalmente una o più librerie per salvare file e gestire tabelle (SQL) o "documenti" (noSQL)
- il runtime o "ambiente di esecuzione": interprete, gestione memoria, librerie (es. JVM, V8, ecc.)
- il middleware o "strato di comunicazione"
- il sistema operativo (Windows, Linux, MacOs)

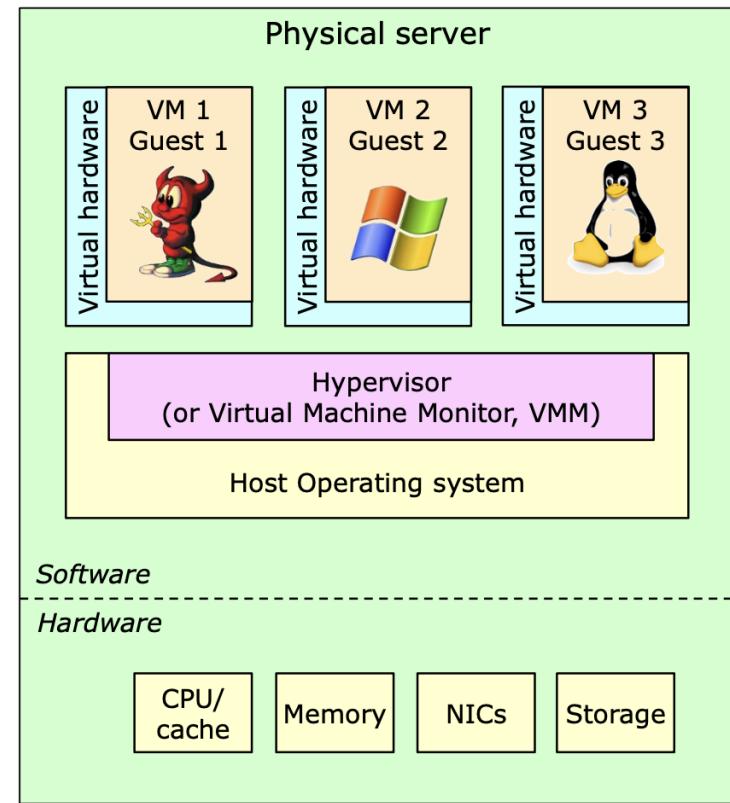


# Virtualizzazione

Un singolo server HW è "sprecato" per una singola applicazione / servizio (es. sito web).

Far coesistere più applicazioni/servizi diventa un problema di *sicurezza* e di *gestione* (chi ha i permessi di fare cosa ed eventuali conflitti software - librerie, runtime, ecc.).

Tramite la **virtualizzazione** si creano più **macchine virtuali** (VM) sullo stesso server HW, ognuna con il suo sistema operativo e le sue applicazioni, una VM per ogni diverso *cliente*.

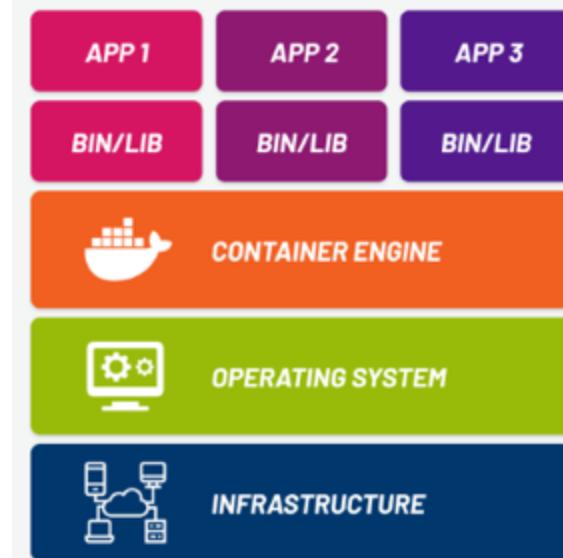


# Containerizzazione

La virtualizzazione si spinge ancora oltre per permettere una più elevata *indipendenza* delle applicazioni da "ciò che sta sotto".

Con i **container** si possono virtualizzare le singole applicazioni o gruppi di applicazioni (con insieme librerie e - micro - sistema operativo).

In questo modo si arriva al punto "ideale" di avere in un container un OS con una singola APP che si può gestire e "traslocare" indipendentemente da tutto il resto.

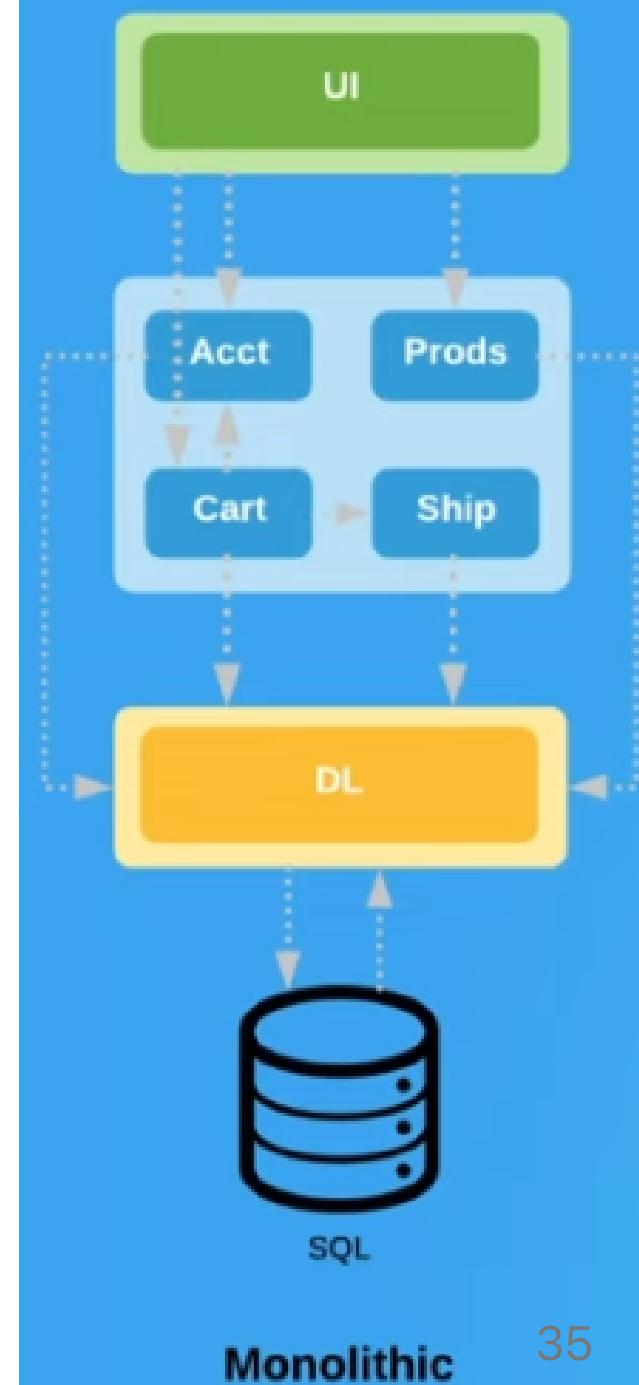


# Struttura di una applicazione

Una applicazione software è composta da:

- interfaccia utente (user interface), comunemente l'interfaccia grafica con cui operiamo, come utenti, su una applicazione
- logica applicativa (business logic), l'insieme di algoritmi, regole, e altro con cui "ragiona" una applicazione
- gestione dei dati (data abstraction layer), la "vista" sui dati e l'interazione con gli stessi

Esempio: sw di videoscrittura, il suo documento, in ambiente C# o Java, che interagisce con disco / sistema operativo / mouse e tastiera



# Applicazioni multi-tier distribuite

Le applicazioni moderne non sono più *monolitiche* (dove UI, BL e DL appartengono alla stessa applicazione) ma *multi-tier* e *distribuite* con in genere una (o più applicazioni) per ogni "tier":

- una per l'UI (es. Web)
- una per la BL (es. Java)
- una per il DL (es. DBMS)

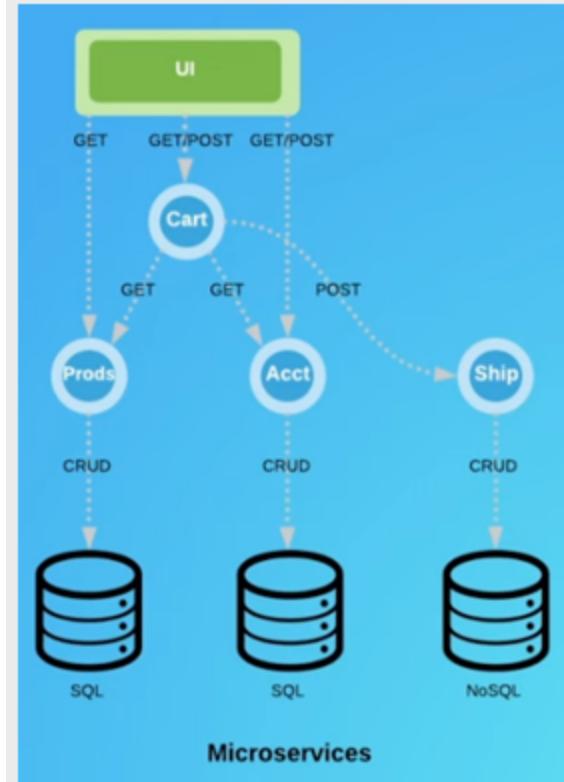
Vantaggi: rendendo i *tier* indipendenti ogni applicazione può essere sviluppata nel linguaggio più adatto, essere aggiornata e "messa in campo" indipendentemente dalle altre, essere gestita e sviluppata da un team separato.

# Microservizi

Per una maggiore libertà l'approccio multi-tier è esteso fino alla generazione di molteplici applicazioni ognuna specializzata in un *micro-servizio*: login/account, carrello, catalogo prodotti, pagamenti.

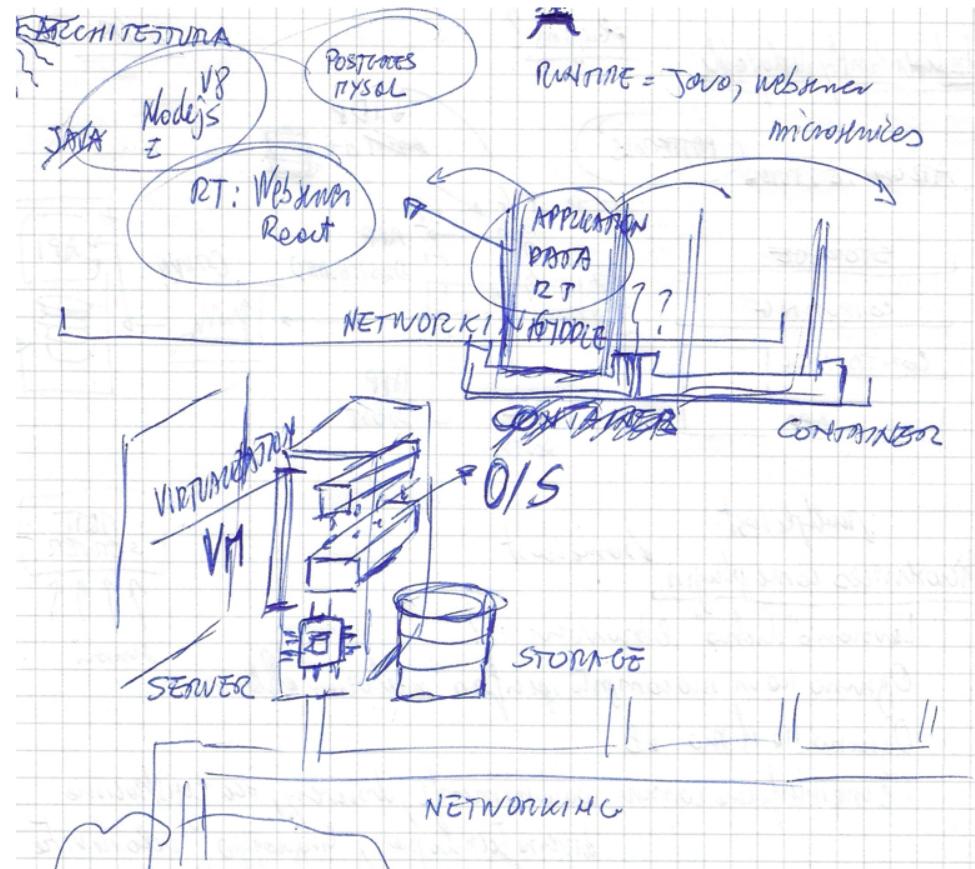
Ogni *microservizio*

- è integrato nello stesso *frontend* (UI),
- ha il proprio database,
- è sviluppato nel linguaggio/ambiente "preferito"
- è aggiornato, deployato, gestito indipendentemente
- comunica tramite API (interfaccia)



# Architettura generale

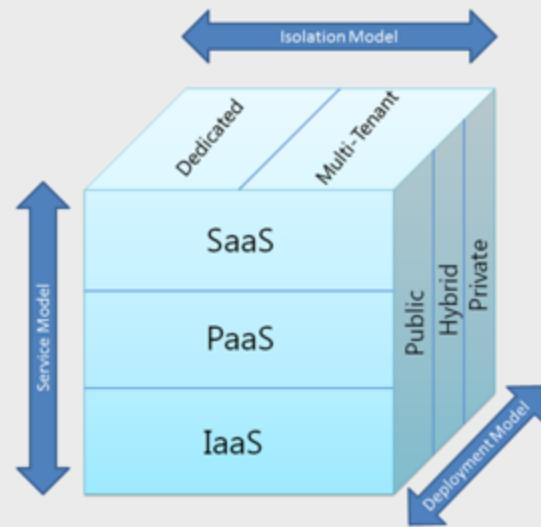
Miro Dashboard



# Modelli di servizio

Detti anche "service models" sono ortogonali ai "deployment models" e agli "isolation models" e definiscono il livello dello stack di sistema e applicativo al di sopra del quale la gestione è a carico del client (e al di sotto del quale è a carico del provider).

Li distinguiamo come "Something as a Service (aaS)": hardware, infrastruttura, piattaforma, software.



## Hardware (aaS)

Dei server reali in uno spazio affittato dove sono garantite "facilities" come alimentazione, connettività, condizionamento.

- simile a colocation (ma l'hardware è di terzi)
- necessario se ci sono grossi vincoli di sicurezza (e.g. banche)

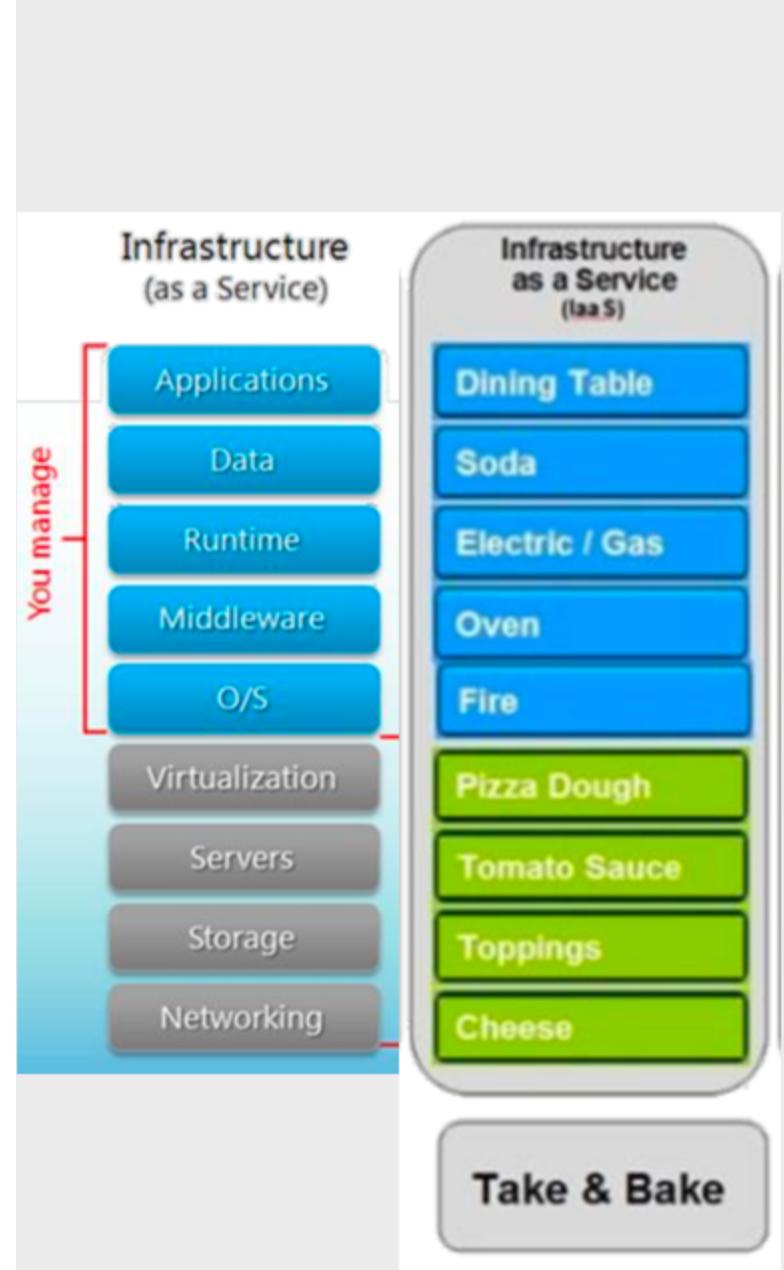
Nota. Più frequente di quanto si immagini (es. se un grosso cloud provider vuole avere la presenza in una zona, ma non è economicamente o burocraticamente sostenibile "aprire" un suo datacenter)

# Infrastructure (aaS)

Dei "server" virtualizzati (e basta)

- a parte l'hardware si gestisce tutto l'aspetto software
  - scalabilità, fail tolerance, ecc.
- indicato se si vuole il controllo completo
- elevate competenze tecniche
- elevata complessità di gestione
- praticamente no lock-in (migro vm o container) perchè gestisco tutto io
- talvolta l'unico modo di migrare in cloud applicazioni legacy (vecchie)

[1] "[IaaS](#)", Manuale abilitazione cloud



## Container (aaS)

Dei "container" virtualizzati

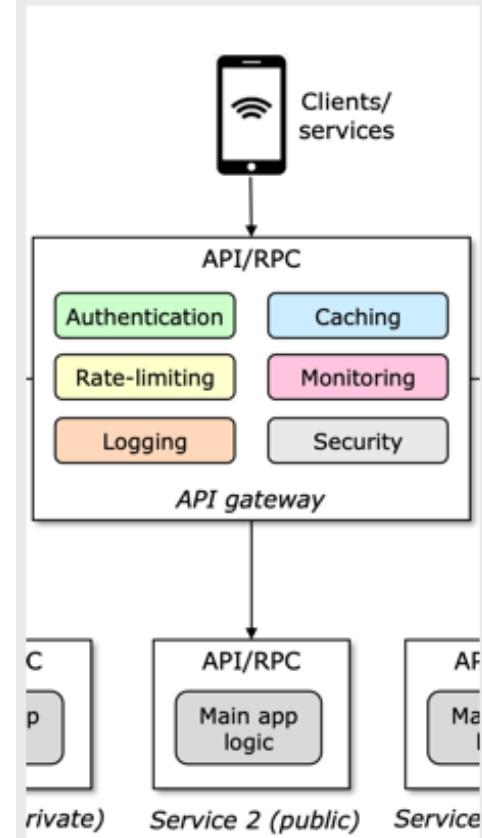
Rimane la "libertà" e flessibilità del modello IaaS, ma un po' più di astrazione (che non è male)

- si orchestrano container (virtualizzazione "leggera") invece di macchine virtuali
- rimane la necessità di elevate competenze ma anche lo svincolo dal lock-in

# Ecosistema Cloud

Numerosi altri aspetti devono essere gestiti in un "cloud" efficace ed efficiente:

- scalabilità: aumentare il numero di applicazioni che gestiscono un servizio quando aumenta il numero delle richieste
- bilanciamento: indirizzare le richieste verso l'istanza (dell'applicazione) più scarica in quel momento
- monitoraggio: raccogliere informazioni sul numero di richieste, tempo di risposta, carico dei sistemi, eventuali interruzioni o malfunzionamenti
- fail tolerance: mantenere repliche delle applicazioni "latenti", ma sincronizzate, in modo da intervenire in caso di malfunzionamenti
- autenticazione, caching, logging, rate-limiting

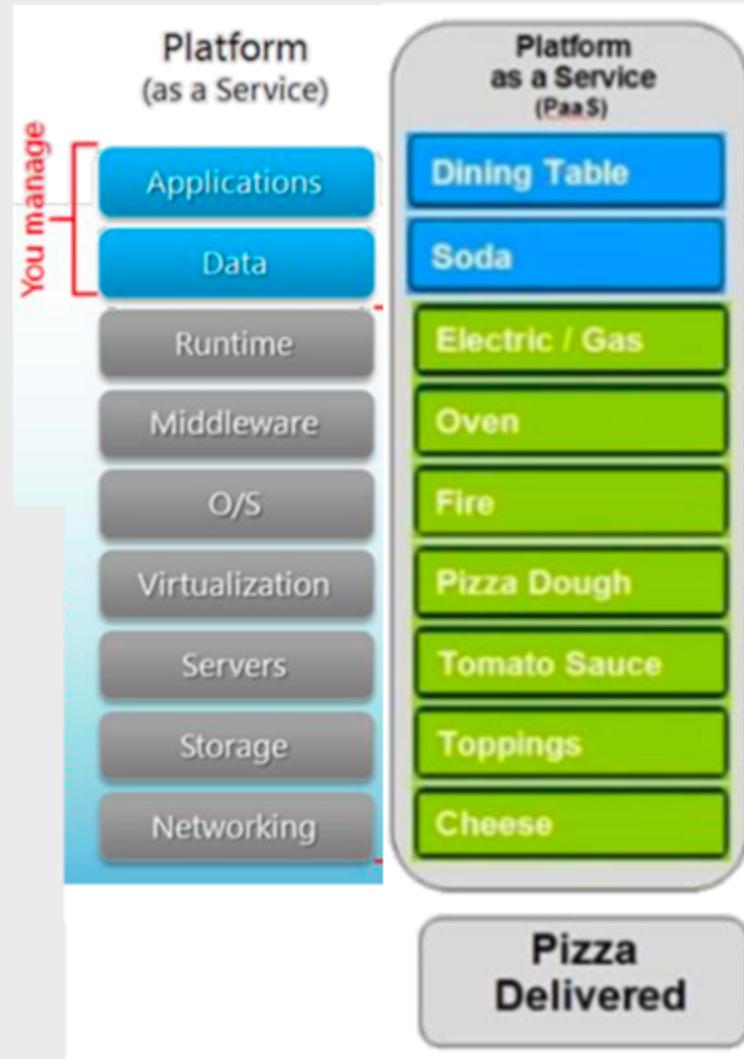


# Platform (aaS)

Delle "piattaforme tecnologiche" (sistema operativo e stack software) in cloud dove sviluppare ed eseguire applicazioni e dati.

- il provider offre servizi "cloud" come database, messaggistica, autenticazione
- il provider gestisce lui l'orchestrazione per scalabilità e fault tolerance
- l'applicazione "rimane" proprietà e competenza del cliente
- rischio di lock-in perchè i servizi non sono standard e variano da provider a provider

[1] "[PaaS](#)", Manuale abilitazione cloud

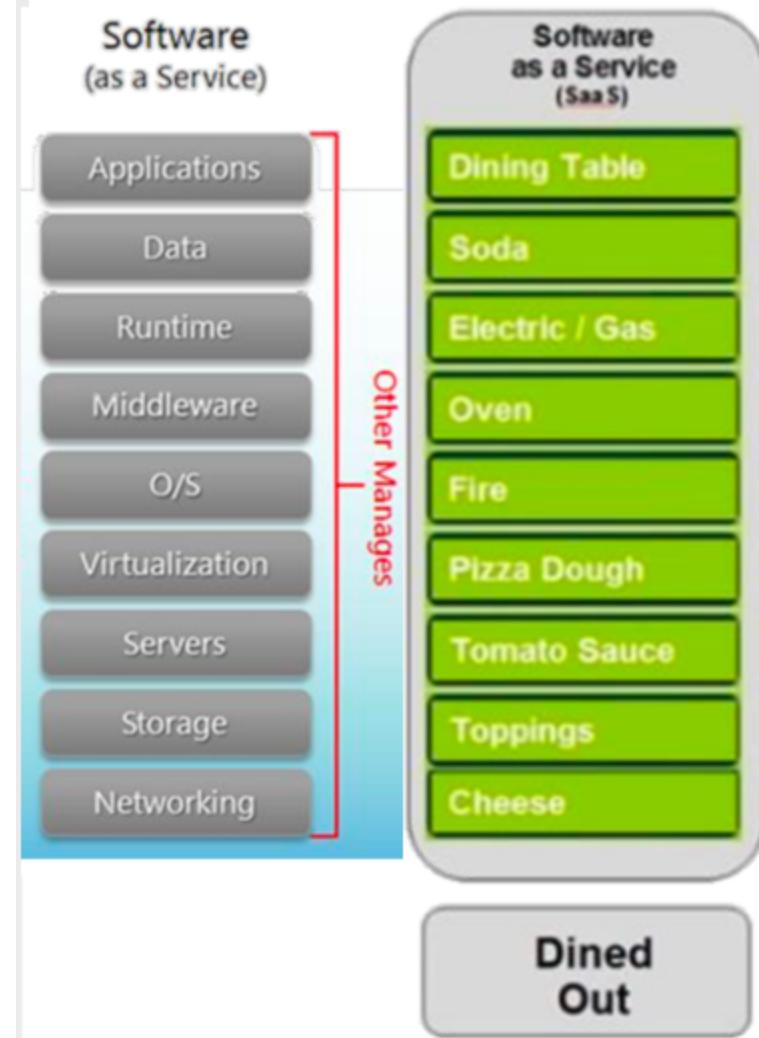


# Software (aaS)

Delle applicazioni fatte e finite, senza bisogno di fare nulla se non pagare (ed esserne vincolati)

- largamente utilizzato per applicazioni "standard" (sito web, email, video streaming)
- elevato rischio di lock-in (verificare sempre di poter esportare/importare i dati per migrare, non necessariamente senza interruzione del servizio, altrove)
- facile e veloce perchè non c'è nulla da gestire

[1] "[SaaS](#)", Manuale abilitazione cloud





## Migrazione al cloud

---

- Assessment di servizi e infrastruttura
- Pianificazione della migrazione: "strategie"
- Migrazione degli applicativi e dei dati
- Verifica e monitoraggio

Rif. "["Manuale di Abilitazione al Cloud - \(MAC\)"](#)", AGID, 2022

---

# Roadmap di migrazione

Il percorso di migrazione è stato definito da AGID tramite 5 macro obiettivi, 9 attività, 6 strategie di migrazione e 4 livelli di priorità.



E' fondamentale procedere in modo iterativo ed incrementale ("farsi le ossa") partendo dagli applicativi che traggono un beneficio significativo dall'adozione del paradigma cloud, che al contempo rappresentano un rischio ridotto per la continua erogazione dei servizi supportati e che risultano relativamente semplici da migrare.

## 1. Lista degli applicativi e dei servizi

- il nome e la versione, l'interlocutore con cui si agisce per gestire lo sviluppo e gli aggiornamenti, eventuali licenze associate;
- i servizi a cui è collegato e le persone di riferimento per tali servizi

## 2. Prioritizzazione

1. *opportunità da cogliere*: applicativi da aggiornare (sw o hw) o il cui mantenimento locale è molto oneroso e no problemi di sicurezza o privacy
2. *rischio minimo*: pochi utenti (magari solo interni) e non necessaria 24/7, ben documentata e conosciuta dal team IT, pochi requisiti normativi
3. *semplice da migrare*: moderni, fornitore attivo e disponibile, no dati critici
4. *altro*: scarsi benefici, complessi, non interrompibili

## 3. Assessment dell'applicativo

Il MAC prevede delle **schede operative** per valutare in dettaglio le caratteristiche di ciascun applicativo.

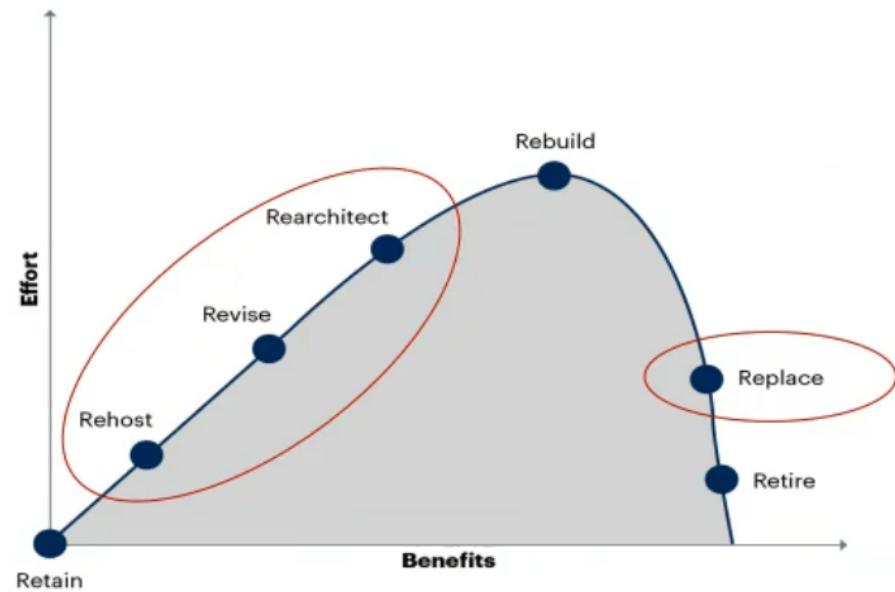
- Aspetti tecnologici: stack tecnologico, componenti sostituibili con equivalente cloud, dimensionamento (cpu, mem, disco), dipendenze, sicurezza
- Vincoli tecnologici: modificabilità del sorgente, disponibilità documentazione, connettività necessaria
- Dati: dimensione, frequenza di consultazione e aggiornamento, "durata", applicativi con cui i dati devono essere sincronizzati, banda di rete
- Parti interessate: rappresentanti di aree, processi impattati
- Bisogni: numero di utenti giornalieri (min, med, max), fasce di utilizzo, licenze, conformità normativa
- Mercato: alternative SaaS con import/export standard dei dati

## 4. Strategie di migrazione

Le principali strategie per la migrazione di applicativi al cloud sono note come le 6R (re-tain, re-host, re-vise, re-architect, re-place, re-tire [e re-build])

Ogni strategia si differenzia per valore generato / *benefit* (risparmio costi infrastruttura e gestione, agilità operazioni configurazione approvvigionamento, scalabilità) e risorse necessarie alla sua realizzazione / *effort*.

La scelta dipende anche dalle risorse disponibili e dal valore strategico dell'applicazione.



## Strategia Re-tain

Decisione di non migrare in cloud in attesa di rivalutare la decisione in futuro.

### Criteri

- forte dipendenza dalla connettività (incompatibilità con l'uso tramite Internet)
- recente acquisizione con quota di ammortamento importante
- basso impatto sulla strategia dell'ente
- forte obsolescenza tecnologica
- assenza di alternativa SaaS
- requisiti di sicurezza e privacy

## Strategia Re-tire

Dismettere gradualmente gli applicativi che non sono più utili dopo averne verificato gli utilizzi da parte di operatori interni o dipendenze verso altri sistemi. Probabile necessità di ri-adattamento dei processi interni.

**Criteri:** applicativi mantenuti per rendere possibile l'accesso e la reportistica di dati storici, che non generano più dati, la cui dismissione è stata procrastinata.

## Strategia Re-host / Lift & Shift

Migrazione in cloud dell'infrastruttura già esistente (applicazioni e dati), senza necessità di reingegnerizzare le applicazioni, praticamente sfruttando "solo" l'hosting.

**Vantaggi.** IaaS con guadagno in scalabilità verticale (poter incrementare le risorse di CPU, RAM e storage) maggiore disponibilità e gestione dei backup remoti.

**Svantaggi.** Non sfrutta appeno le potenzialità cloud e "invecchia" velocemente (meglio limitare a servizi, quasi, in dismissione).

## Strategia Re-vise (re-platform)

Migrazione al cloud con sostituzione / adattamento di alcune componenti per meglio sfruttare le funzionalità cloud.

**Vantaggi.** CaaS con database, bilanciatori di carico, runtime come container, firewall, ecc.

**Svantaggi.** Con tempo, risorse, interesse si può fare di meglio.

## Strategia Re-architect

Ripensare significativamente l'architettura core dell'applicativo in ottica cloud per adottare appieno i servizi cloud-native (es. API gateway, microservizi con autoscaling) e massimizzarne i benefici.

**Vantaggi.** Elevata disponibilità, scalabilità, osservabilità, resilienza, provisioning delle risorse, deployment indipendente dei componenti, monitoring.

**Svantaggi.** Conoscenza tecnica elevata (non facile da trovare) e specifica per quel provider, tempi di sviluppo, lock-in significativo.

Per applicazioni strategiche dell'ente con elevato carico di utilizzo e di cui si dispone del know-how.

## Strategia Re-place / Re-purchase

Rimpiazzare un applicativo installato e gestito on-premise con la controparte SaaS.

**Vantaggi.** Migrazione immediata con minori costi iniziali ed eliminazione dell'infrastruttura a supporto e dell'attività di gestione.

**Svantaggi.** Total cost of ownership potrebbe essere maggiore sul lungo periodo, lock-in (limitato, perché in genere sono servizi standard), integrazione con servizi on-premise (es. autenticazione), assenza di controllo in caso di down.

Per applicazioni "standard" (email, disco condiviso, sito web)

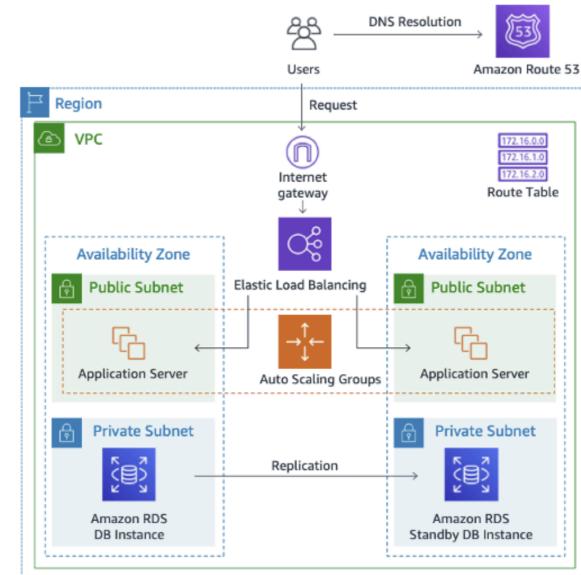
## 5. Analisi costi - benefici

1. definizione del periodo temporale su cui calcolare il ritorno sull'investimento (tipicamente 5 anni)
2. verifica dei costi attuali dell'infrastruttura e loro proiezione sul periodo temporale
3. stima dei costi dell'infrastruttura cloud e loro proiezione sul periodo temporale
4. stima dei costi di migrazione
5. stima dei costi di post-migrazione
6. valutazione dei costi rispetto ai benefici tangibili ed intangibili

# Calcolatore dei costi del provider

Esempio: [calcolatore AWS](#)

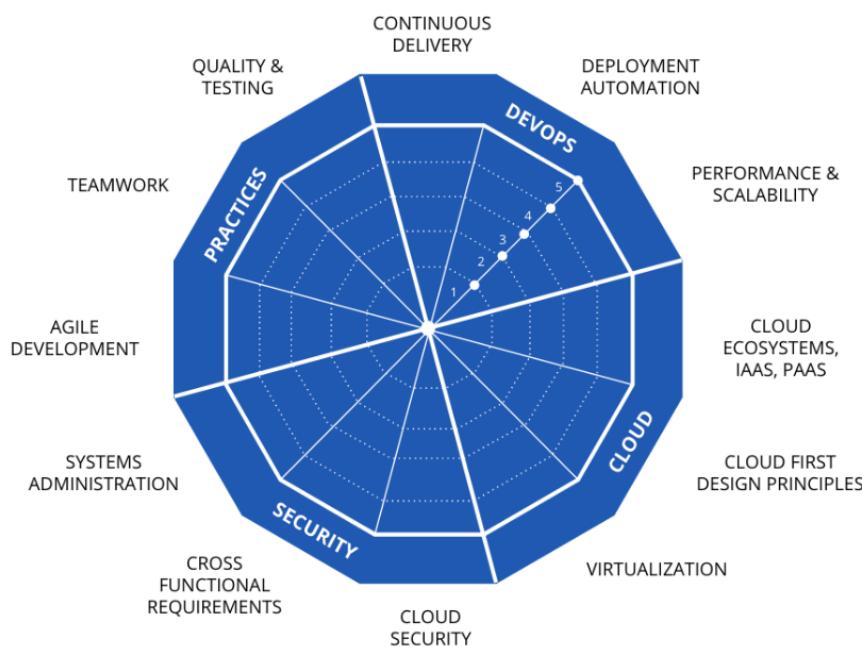
- decine di servizi
- EC2 (i.e., computing)
  - cpu, mem, disco
  - tariffazione del traffico (inbound gratis, outbound a pagamento)
  - eventuale free tier
- servizi aggiuntivi: database, notifiche (SNS), archiviazione (S3), firewall, sviluppo



Esempio: applicazione web con autoscaling e database replicato su due zone.

## 6. Competenze

Occorre individuare le competenze presenti internamente e quelle mancanti. Il MAC mette a disposizione uno **strumento per la valutazione delle competenze** che permette di definire uno "spider-chart"



**Dreyfus Model**

Assessment	
<b>5</b>	Esperto
	Abbandona le regole a favore dell'esperienza e dell'intuizione che permettono un approccio più efficiente e orientato all'obiettivo.
<b>4</b>	Competente
	Regole come fallback. Sviluppo di una visione di insieme e capacità di fare scelte sulla base del contesto e dell'esperienza.
<b>3</b>	Intermedio
	Conoscenza sufficientemente profonda delle regole associate alla skill e capacità raggiungere obiettivi concreti autonomamente.
<b>2</b>	Principiante avanzato
	Maggiore autonomia nell'applicazione della skill in maniera situazionale e sempre tramite il supporto di regole.
<b>1</b>	Principiante
	Nessuna o pochissima conoscenza. Necessita di istruzioni e tutorial precisi per essere produttivo.

## Le altre fasi di migrazione

Le altre fasi, più di carattere manageriale, possono essere direttamente approfondite sul MAC

7. Pianificazione della migrazione
8. Esecuzione della migrazione degli applicativi e dei dati
9. Validazione: dopo la migrazione

# Approfondimenti

- Materiale del corso di "[Cloud Computing](#)" del prof. Risso presso il Politecnico di Torino, 2023
- [Manuale di Abilitazione al Cloud](#), AGID, 2022
- [Polo Strategico Nazionale](#)
- [Programma di Abilitazione al Cloud](#) con kit e framework.