

Cloud Computing and Networking

Introduzione al Cloud Computing

- **Obiettivo principale del cloud computing:** fornire risorse hardware e software tramite Internet, evitando gli alti costi di costruzione e implementazione da zero
 - Basato sul modello **pay-as-you-go** (paga per l'uso effettivo)
 - Combina elementi di **grid computing e utility computing**, riducendo costi e necessità di infrastrutture proprie
 - Il termine cloud deriva dal simbolo usato dagli ingegneri di rete per rappresentare Internet, a indicare astrazione e complessità nascosta
-

Modelli di Computing Tradizionali

- **Monolitico Centralizzato**
 - Un unico sistema centrale gestisce tutto
 - Combina logica di presentazione, business e dati in un solo programma su una macchina fisica
 - Adatto per applicazioni semplici, non per accesso simultaneo da parte di molti utenti
 - **Peer-to-Peer (P2P)**
 - Rete in cui ogni PC agisce sia da client che da server
 - Ideale per piccole reti (fino a ~10 peer) per la condivisione file e stampanti
 - Vantaggio: basso costo; svantaggi: prestazioni limitate, sicurezza e backup complessi
 - **Client-Server**
 - Configurazione master-slave con un server dedicato e più client
 - **Two-Tier:** presentazione sul client, dati sul server; semplice ma poco scalabile
 - **Three-Tier:** separa presentazione, logica business e dati su tre livelli; migliora scalabilità e affidabilità
 - **N-Layer Three Tier Model Distribuiti:** più livelli su macchine diverse; fa uso di Web Services per interoperabilità
-

Modelli di Computing ad Alte Prestazioni

- **Cluster Computing**

- Insieme di computer connessi in rete ad alta velocità che lavorano come un sistema unico.
- Vantaggi: performance, alta disponibilità, bilanciamento carico e tolleranza ai guasti.
- Tipi:

- **Load-Balancing Cluster** (scaling orizzontale)

Obiettivo è quello di distribuire richieste clienti a differenti Compute Node attivi. Assicura una gestione efficiente di un numero enorme di richieste clienti. Tutti i Compute Node eseguono lo stesso tipo di servizi simultaneamente. La funzionalità di questa rete è DISTRIBUIRE le richieste tra i nodi attivi.

Si può scalare orizzontalmente aggiungendo Compute Node nella rete.

Le componenti di questa rete sono:

- Bilanciamento di Carico Software Based: algoritmi per distribuire il carico.
 - Bilanciamento di Carico Hardware Based: usa hardware specializzato per la distribuzione.

- **High-Availability Cluster** (failover)

Assicura servizio continuo con minimo caricamento. Sostanzialmente se un nodo fallisce, vogliamo poter comunque avere il servizio disponibile.

Meccanismo di Fail-Over: un nodo passivo fa ripartire app del nodo fallito. Per capire se un nodo è fallito veramente si utilizza:

- Interface Link Method: il nodo passivo rileva un fallimento.
 - Heart Beat Method: il nodo attivo manda un segnale regolare, se non viene più ricevuto dal passivo esso fa ripartire le applicazioni

Il nodo può essere:

- attivo: serve le richieste dei clienti.
 - passivo: sostituisce un nodo nel caso di fail.

- **High-Performance Cluster** (compute farm per calcoli scientifici)

serve per quelle applicazioni dove il tempo di risposta è cruciale, si ha un aumento del throughput.

Si ha la scalabilità in verticale (migliora la potenza del nodo nel cluster) o in orizzontale (si aggiungono nodi)

Si ha due funzionalità di questa tipologia di Cluster:

- Compute Farm: i lavori sono dati non ad una singola macchina ma ad un cluster di macchine. Il Cluster gestisce le risorse e assegna i jobs in una coda di lavori. I jobs possono essere fatti eseguire simultaneamente o aspettare nella coda per la disponibilità di risorse.
 - Middleware Cluster: cerca delle risorse disponibili e assegna il lavoro, mantiene il flusso del lavoro, combina i risultati e li fa ritornare all'utente.

- **Fault-Tolerant Cluster** (esecuzione in lockstep per zero downtime):
assicura zero caricamento con un servizio continuo. Si hanno dei nodi con lo stesso servizio simultaneamente, se il primo sistema fallisce il secondo prende il controllo immediatamente. Sistema Ridondante.

- **Grid Computing**

- Condivisione di risorse distribuite geograficamente tra organizzazioni.
- Nodi eterogenei (differenti hardware), coordinati come un'entità virtuale (combina le risorse in un singolo sistema virtuale).
- I partner condividono tra di loro risorse per un obiettivo comune.
- Applicazioni scientifiche; limiti: mancanza standard, complessità software, non interattivo.

Vediamo come è fatto, l'infrastruttura è formata da: Coordinazione della Grid (assicura la gestione delle risorse e integrazione), Politiche Amministrative (Differenti Collaboratori con varie politiche).

Le componenti effettive sono:

- Interfaccia utente: portale web per accedere alla griglia di risorse.
- Infrastruttura Grid: collezione di rete eterogenea di risorse.
- Grid Server: coordina e gestisce una rete eterogenea di risorse.

Componenti della Grid Server: si hanno due componenti:

- Grid Portal: autentica gli accessi dell'utente.
- Componente Broker: scopre e identifica la risorsa richiesta utilizzando **#MDS** (Monitoring and Discovery Service).

- Componenti della Grid Server: si hanno due componenti:

- Grid Portal: autentica gli accessi dell'utente.
- Componente Broker: scopre e identifica la risorsa richiesta utilizzando **#MDS** (Monitoring and Discovery Service).

- Scheduler: schedula i lavori degli utenti da runnare nella griglia e muove i task nella griglia ad un tempo schedulabile.
- Gestione Risorse e Data: è formata dai seguenti componenti
 - **#GASS** (Grid Access to Secondary Storage): gestisce i dati durante l'esecuzione di un task.
 - **#GRAM** (Grid Resource Allocation Manager): gestisce risorse durante l'esecuzione di un task.
- Differenze chiave:
 - Grid: focalizzato sulla collaborazione scientifica, non ha modello di business pubblico.
 - Cloud: orientato a servizi on-demand economici e scalabili.

Modelli di Computing per la Vendita di Risorse IT

- **Utility Computing**
 - Fornisce risorse come un'utenza domestica, con fatturazione basata sull'uso.
 - Inserito all'interno del paradigma cloud
-

Motivazioni e Definizioni

- **Pionieri**
 - Salesforce: applicazioni come utilità via Internet
 - Amazon: ha creato S3 ed EC2, aprendo la strada al pay-per-use
 - Google App Engine (2008), Microsoft Azure (2009)
- **Definizione sintetica:** modello di calcolo via Internet, con risorse e prezzi pay-per-use e adattamento dinamico alle esigenze

Il modello di Cloud Computing ha l'obiettivo principale di provvedere ad hardware e software tramite Internet. Usando una politica di Pay-As-You-Go, ovvero, gli utenti pagano quello che stanno usando.

La differenza tra Cloud e Grid:

- Cloud: provvede a risorse su richiesta da utenti, crea supporti per infrastrutture IT velocemente.
 - Grid Computing: condivisione di risorse tra partner, supporta collaborazioni ma non è aperta al pubblico la rete. Spesso utilizzato per ricerche.
-

Perchè Cloud Computing?

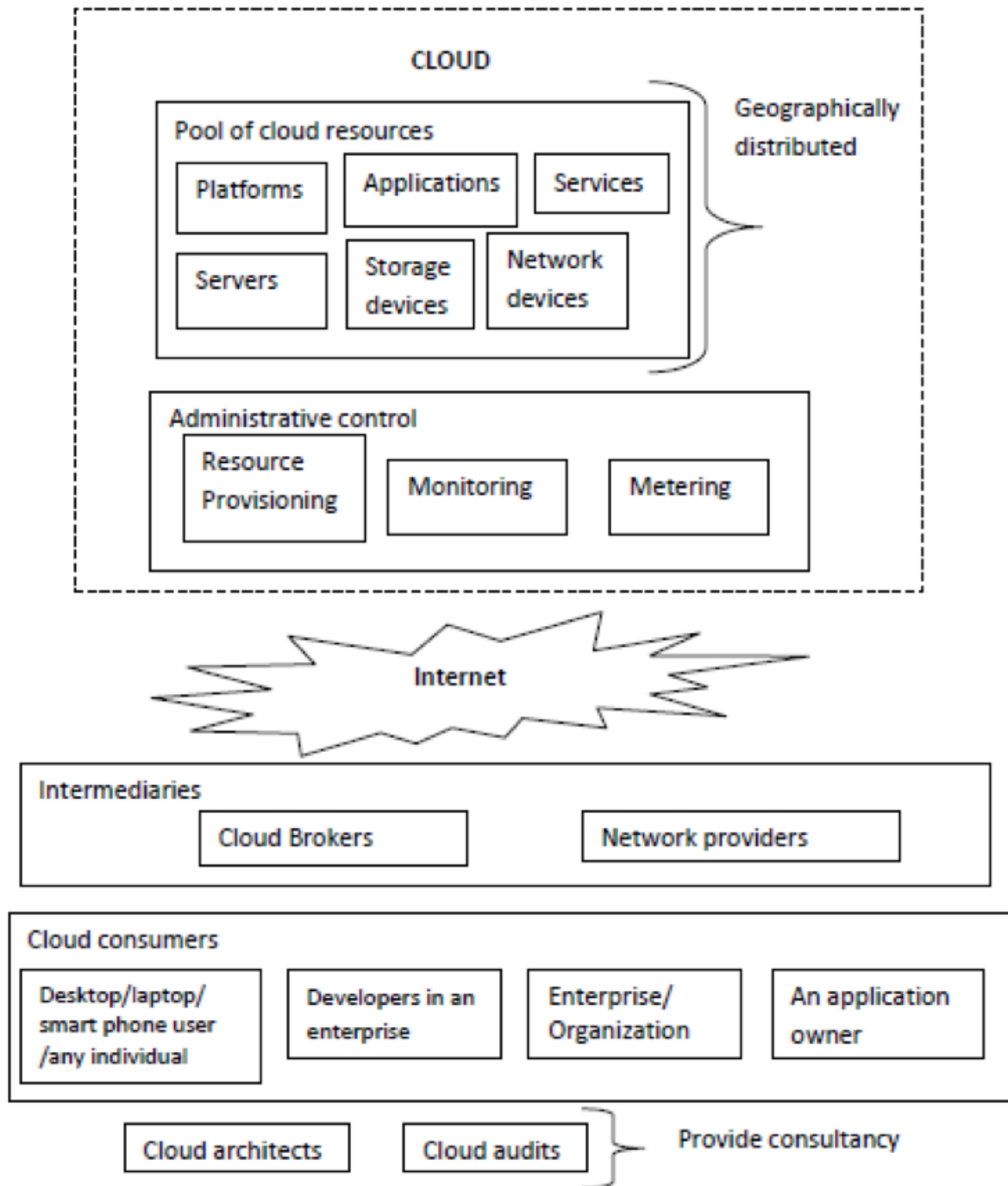
Perchè riflette il fatto che i consumatori non sanno i dettagli precisi dell'infrastruttura dentro un cloud (come una nuvola, non si ha cosa si cela dentro).

Le risorse sono: Servers, dispositivi di memoria, dispositivi di rete, piattaforme, applicazioni software.

Stakeholder nel Cloud Computing

1. **Cloud Service Providers:** gestiscono infrastrutture e servizi cloud
2. **Cloud Consumers:** utilizzatori finali (privati, sviluppatori, aziende)

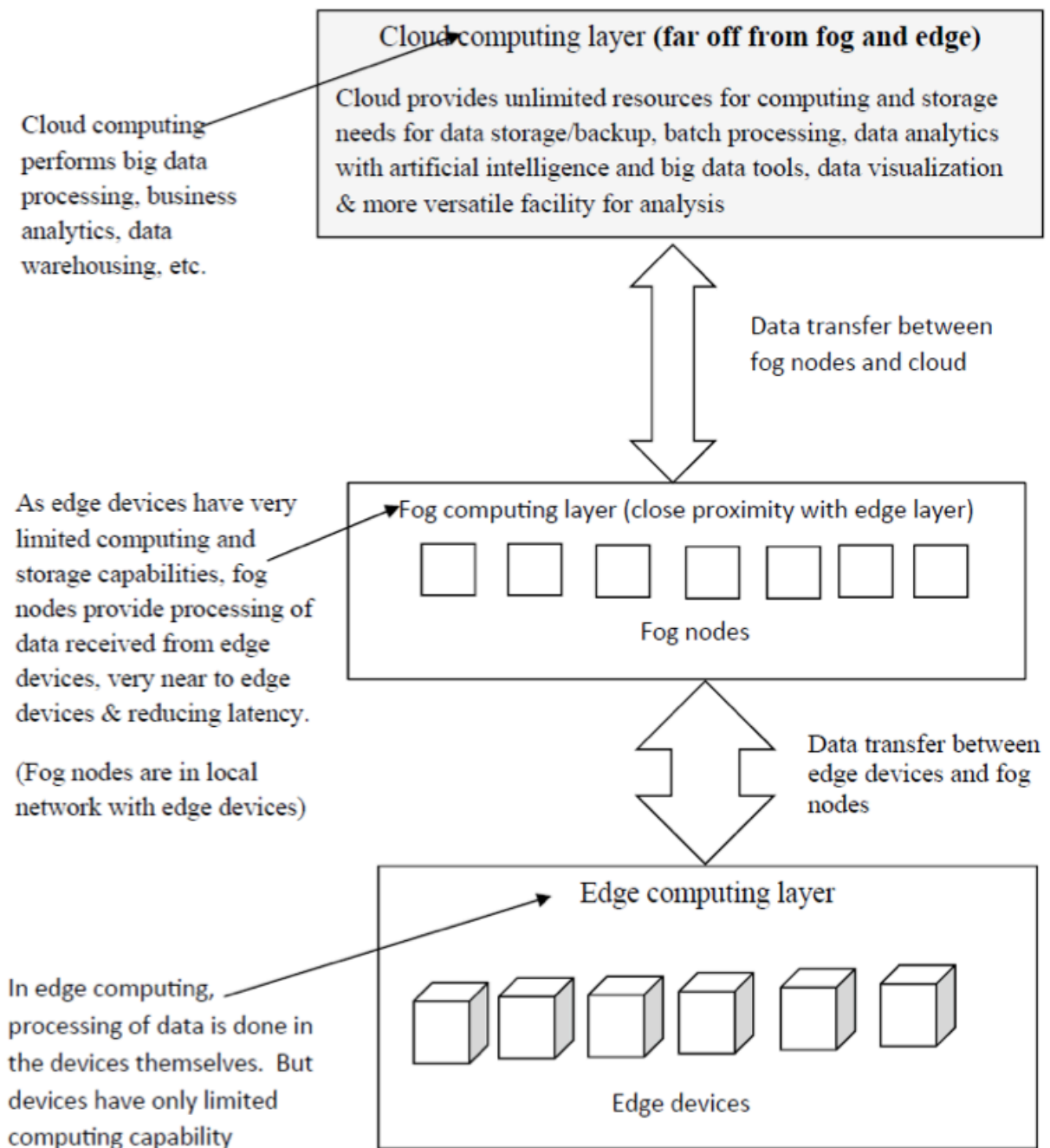
3. **Cloud Brokers:** intermediari (per fatturazione, selezione fornitori e integrazione) tra i provider di servizi e i consumatori. Permettono di identificare il giusto provider del servizio, integra altre tipologie di servizio, da supporto.
4. **Cloud Architects:** progettano soluzioni cloud personalizzate
5. **Cloud Auditors:** valutano sicurezza, prestazioni e conformità



Paradigmi Recenti: Edge, Fog, Cloud

- Edge Computing

- Elabora dati vicino ai sensori; utile per IoT
- Produzione di insight in tempo reale, bassa latenza, basso costo
- Processa i dati nel punto di acquisizione.
- **Fog Computing**
 - Introduzione di livello intermedio tra edge e cloud (Cisco, 2014)
 - Riduce traffico verso il cloud e mantiene qualità del servizio
 - Processa i dati tra i dispositivi edge e i cloud , riducendo il flusso della rete e mantenendo un certo QoS.
 - Riceve i dati dai dispositivi edge, li analizza e vede quali mandare al cloud.
- **Cloud Computing**
 - Adatto a elaborazione batch e analisi su larga scala.
 - Scenario dove serve tanta capacità di calcolo e memoria.
 - Offre storage e calcolo ampi ma con maggiore latenza e costi più alti.
- Ogni paradigma serve esigenze specifiche; non si sostituiscono a vicenda totalmente. Se voglio che un sensore elabori velocemente dei dati con calcoli semplici non conviene portarlo al cloud!



Altri esempi:

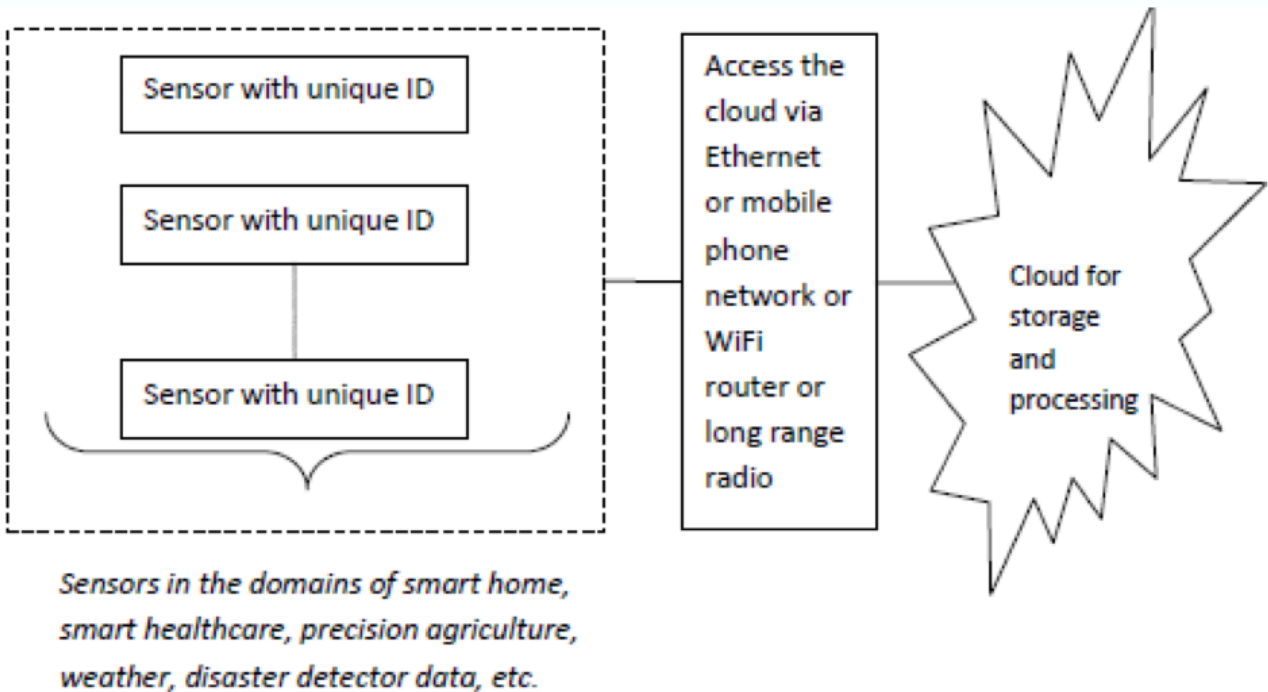
Examples:

- Edge Computing:
 - Application: Security and surveillance in airports.
 - Function: Real-time analytics with smart cameras equipped with processing intelligence to detect hijackers and take immediate action.
- Fog Computing:
 - Application: Car-to-car traffic communication.
 - Function: Local network of nodes analyzes data from various sensors in autonomous cars, providing real-time insights without latency.
- Cloud Computing:
 - Application: Atmospheric and dispersion analysis with weather sensors.
 - Function: Analyzes large volumes of historical data, suitable for big data analytics and AI models, offering unlimited storage and computing resources.

Cloud + IoT / Big Data

L'Internet of Things è una collezione di dispositivi o sensori che hanno degli identificativi univoci.

- **Cloud e IoT:** l'infrastruttura cloud supporta l'archiviazione e analisi dei dati generati dai dispositivi IoT



- **Big Data:** utilizzo di sistemi come HDFS, MapReduce, Apache Spark per elaborare Volume, Velocità e Varietà. Con Varietà si intende il fatto che ci possono essere dati con differenti formato.

Limiti del Cloud Computing

1. Necessità di connettività Internet
2. Possibili interruzioni del servizio
3. Il multi-tenancy (istanza/risorsa tenuta da più sistemi) facilita fuga dati
4. Incertezza sulla collocazione fisica dei dati
5. Problemi di sicurezza, privacy e normativa
6. Possibile mancata eliminazione sicura dei dati
7. SLA non standardizzati
8. Vendor lock-in: differenti provider utilizzano differenti architetture o API, difficile integrare servizi tra venditori o migrare dati.

- 9. Latenza e prestazioni variabili: performance è instabile durante picco di carichi
 - 10. Mancanza di standard comuni: inter operabilità tra cloud limitata
-

Caratteristiche Distintive del Cloud Computing

- **Provisioning on-demand:** risorse attivate automaticamente.

Abbiamo detto che una caratteristica molto importante del Cloud Computing è proprio la scalabilità, ovvero, l'abilità di un sistema di gestire carichi senza compromettere il QoS. Si può fare una scalabilità di due tipi:

- Scale Up (Verticale): si aggiunge le risorse ai nodi già esistenti.
- Scale Out (Orizzontale): si aggiunge più nodi.

Lo si può fare in occasione di periodi con grossi carichi, permettendo una fruibilità del sistema anche in caso di grossi carichi della rete.

Schemi di Approvvigionamento:

- Statico: sono allocate più risorse per gestire i picchi di carico della rete. Questo porta però ad una sotto utilizzazione di queste risorse nei periodi non di picco.
- Dinamico: le risorse sono allocate solo nel momento del bisogno in real time, riduce i costi e assicura un utilizzo efficiente delle risorse.

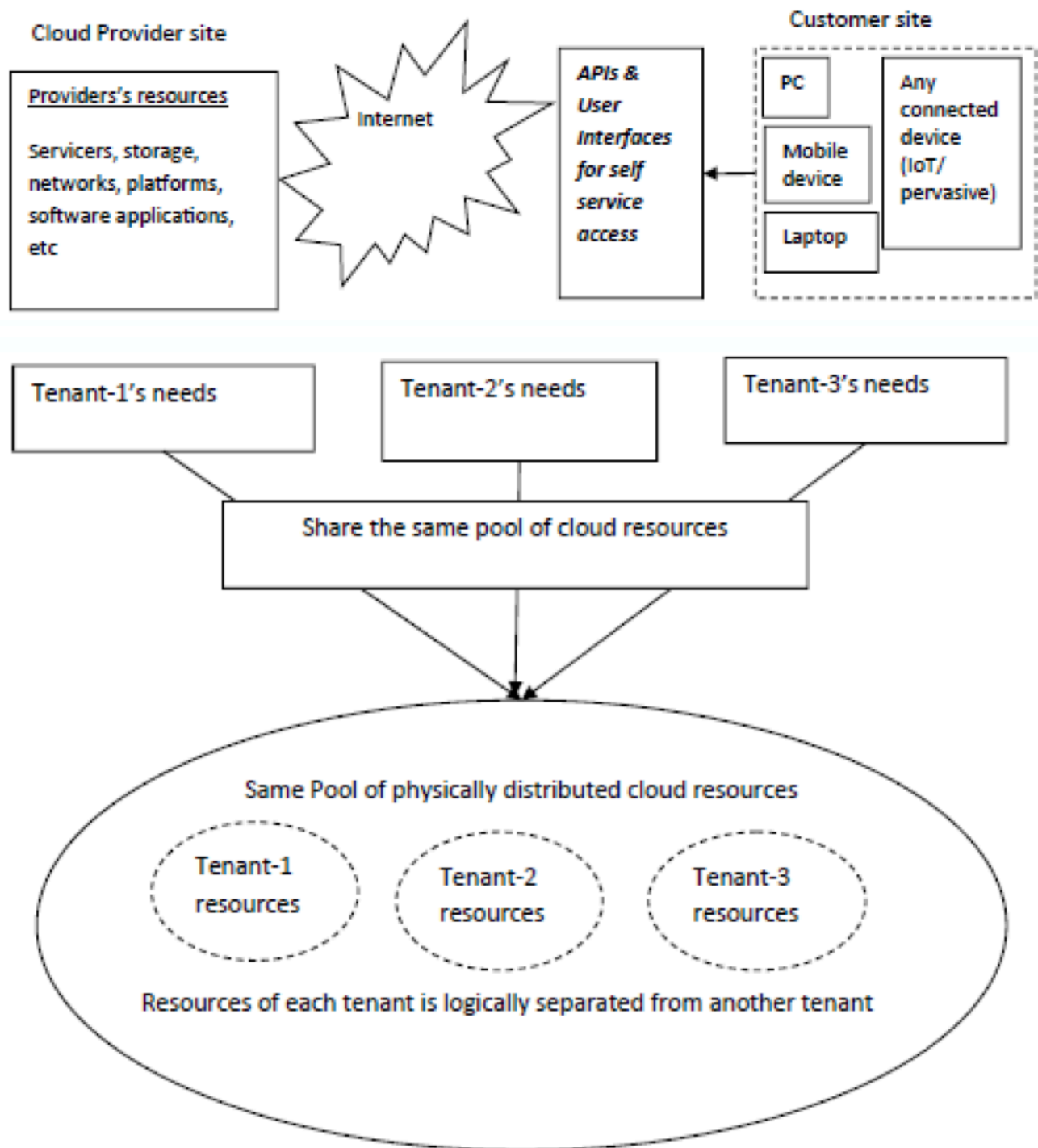
Per gestire questi periodi di carico si hanno due tipologie:

- Variazioni a Lungo Termine: si calcolano queste variazioni con schemi previsionali per anticipare il periodo.
- Variazioni a Corto Termine: schemi reattivi per allocare velocemente.
- **Accesso self-service:** gestione autonoma online delle risorse e dell'accesso da parte degli utenti. Utente può indipendentemente accedere e gestire le risorse del cloud attraverso un interfaccia online senza interagire con il provider.
- **Resource pooling / Multi-tenancy:** risorse condivise, dati separati.

Gli utenti condividono risorse dinamicamente allocati su richiesta.

Si definisce Resource Pooling come un cloud che provvede ad un insieme di risorse computazionali che vengono richieste da più utenti. Mentre il Multi Tenancy condivide la stessa risorsa fisica ma i loro dati e applicazioni sono logicamente separate.

Allocazione Dinamica - Trasparenza(il tenants non sa dove sta risorsa fisica) - Isolazione(i dati tra tenants sono tenuti separati)



- **Rapid elasticity:** scalabilità rapida orizzontale o verticale. L'abilità di scalare velocemente servizi in base a la richiesta dell'utente. Questa capacità deve assicurare minimo delay e viene automatico.
- **Measured service:** monitoraggio e fatturazione basata su utilizzo. Misura le risorse utilizzate dal consumatore, essi infatti sono tassati in base a quanto usano le risorse. Gli schemi dei prezzi sono tramite una iscrizione o in base a quanto effettivamente viene usata la risorsa.
- **Broad network access:** fruizione da vari dispositivi e locazioni.
- **Disponibilità, resilienza, affidabilità:** garantite da ridondanza, backup, monitoraggio.

Architettura del Cloud

Front-End: interfacce utente e API

Back-End: rete, server e servizi, articolati in:

1. **Physical Layer:** infrastruttura hardware fisica
2. **Virtualization Layer:** ipervisore che astrae risorse
3. **Cloud Services Layer:**
 - **IaaS:** infrastruttura virtuale (es. AWS, Azure)
 - **PaaS:** infrastruttura + piattaforma di sviluppo (es. App Engine)
 - **SaaS:** software completo “chiavi in mano” (es. Salesforce)

4. **Scheduling / Provisioning / Pricing Layer:** gestione risorse, fatturazione, sicurezza

Front end Cloud users connect to back end using PC, Laptop, Smart Phone, Ubiquitous device, IoT or any wearable/connected device

Network access for different deployments

Internet &
public APIs

Private LAN &
APIs

Private LAN,
Internet & APIs

Public
cloud

Private
cloud

Hybrid
cloud

Back end (cloud) is deployed as

Back end (Cloud)

Management

Scheduling, provisioning, metering, pricing and
other services

Cloud services

Software as a Service (SaaS)

Platform as a Service (PaaS)

Infrastructure as a Service (IaaS)

Virtualization Layer

Physical Layer

Security

Physical Layer:

Per quanto riguarda il back end si ha il livello fisico che contiene risorse hardware come server, CPU, memoria, dispositivi, dispositivi di rete e altre componenti. Possono essere distribuiti anche in altri posti e non essere confinato in un unico posto.

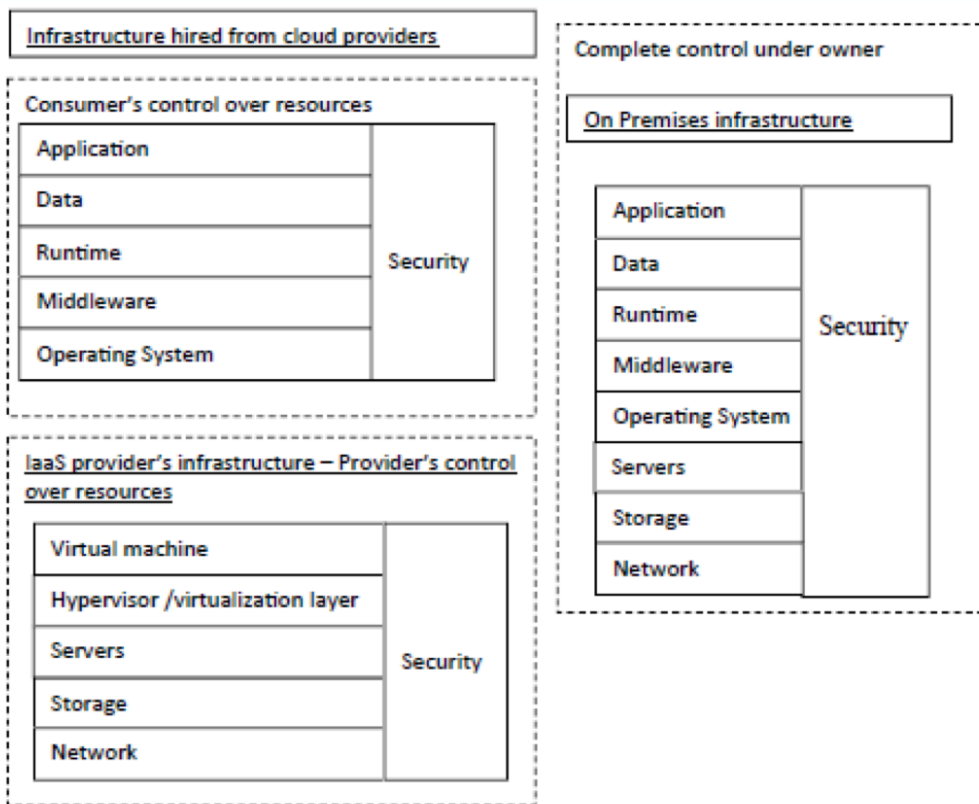
Virtualization Layer:

Questo livello astrae quello che c'è nel livello fisico omogenizzando l'eterogenità, si ha un VMM (Virtual Machine Monitor) e un Hypervisor che combinano tutti i componenti hardware fisici in una visione globale unificata. Inoltre sta attento alla partizione di più risorse per tenants multipli.

Cloud Services Layer:

È composto da tre diverse classi:

- **IaaS (Infrastructure as a Service):** provvede a risorse hardware come memoria server e componenti di rete. È un servizio che provvede con risorse computazionali virtuali, distribuite e condivise. Il provider offre server (hostare app o servizio), memoria (data management), Network(assicurare il data flow), Hypervisor(Software che crea e gestisce le macchine virtuali), Macchine Virtuali. Le responsabilità dell'Utente sono quelle di installare e gestire il sistema operativo, eseguire gli ambienti, i dati e le applicazioni sulle VM provvedute dal provider.



Benefici:

- Elimina costo manutenzione e setting di un data center
- Risponde velocemente per evolvere il business con risorse scalabili.
- Deploy di una nuova infrastruttura in minuti, permettendo il test di nuove idee.
- Accesso Diretto a server, rete e memoria.
- Capacità di recupero in caso di disastri.

- **PaaS (Platform as a Service):**

i provider offrono sia infrastruttura hardware che software per lo sviluppo di un applicazione.

Le componenti offerte sono:

- infrastrutture hardware: server, memoria e networking
- Piattaforme Software: server web, applicazioni server, database server
- Ambienti di sviluppo:python j2EE, .Net

L'utente ha sotto controllo:

- le applicazioni che vengono installate e i dati.

Mentre il provider del PaaS gestisce hardware e piattaforma.

Benefici:

- transizione da desktop in web application
- focus su lo sviluppo senza preoccuparsi dell'esecuzione e della gestione dei carichi.
- meno bisogno di esperti di rete e amministrazione
- elimina per licenza software, riduce i costi associati alla manutenzione

- SaaS :
i provider offrono applicazioni completamente sviluppate agli utenti tramite delle iscrizioni. L'utente quindi può semplicemente accedere e usare l'applicazione data dal SaaS provider. Questa gira sulla infrastruttura del provider. Le responsabilità sono tutte del provider. Viene utilizzato per compagnie non tecniche che hanno bisogno di un software completamente sviluppato. Non si ha bisogno di installazione e costi di manutenzione, accesso remoto su internet da qualsiasi locazione. Utenti possono accedere ogni volta da ogni posto. Gli utenti pagano l'uso.

Scheduling, Provisioning e Pricing Layer:

Per quanto riguarda la componente di scheduling esso alloca le risorse in base alla richiesta.

Il Provisioning provvede a dare le risorse schedulate agli utenti.

Il componente di Monitoring monitora la salute delle componenti hardware.

I servizi di Metering misurano quanto è l'utilizzo delle risorse per il calcolo del prezzo.

Componente di Pricing: tassa i consumatori in base alla sottoscrizione o l'uso dei modelli.

Componenti di Sicurezza: assicura la protezione dei dati ad ogni livello.

Modelli di Deployment del Back End

Public Cloud:

servizi condivisi su Internet pubblico e condiviso da numerose organizzazioni. Scalabile e Economico.

Questo tipo di deployment offre i servizi e l'infrastruttura fuori sede tramite internet per utenti con un modello pay per use.

I servizi sono accessibili tramite API al pubblico, ideale per applicativi che sono accessibili ad un grande numero di persone.

Purtroppo ci sono dei difetti:

- Multi Tenancy: gli utenti devono condividere le stesse infrastrutture fisiche tramite la virtualizzazione.
- Minacce nella Sicurezza: infrastrutture condivise hanno rischi potenziali.

Private Cloud:

dedicato a una singola organizzazione; sicurezza e controllo maggiori. Il cloud è tenuto e gestito da una singola organizzazione e offre servizi solo agli utenti di quella organizzazione su una LAN privata.

Modelli di deployment:

- Within Organization: il cloud privato è sviluppato e gestito internamente all'organizzazione.

- With a Third Party Provider: il cloud privato è messo su e gestito da un terzo provider ma dedicato ad una singola operazione.

Questa tipologia di cloud permette un accesso esclusivo agli utenti e i loro partners dell'azienda possono accedervi. Non si ha il Multi Tenancy, avendo una sicurezza migliore poichè infrastrutture isolate dalle altre aziende, grosso controllo sull'ambiente del cloud.

Hybrid Cloud:

combinazione di pubblico e privato; flessibilità e salvaguardia dei dati sensibili.

Per applicativi, servizi e dati sensibili si utilizza un cloud privato. Per invece utilizzo ad un utente pubblico si utilizza il cloud pubblico sull'internet.

I vantaggi sono la flessibilità essendo sia privato che pubblico, scalabile .

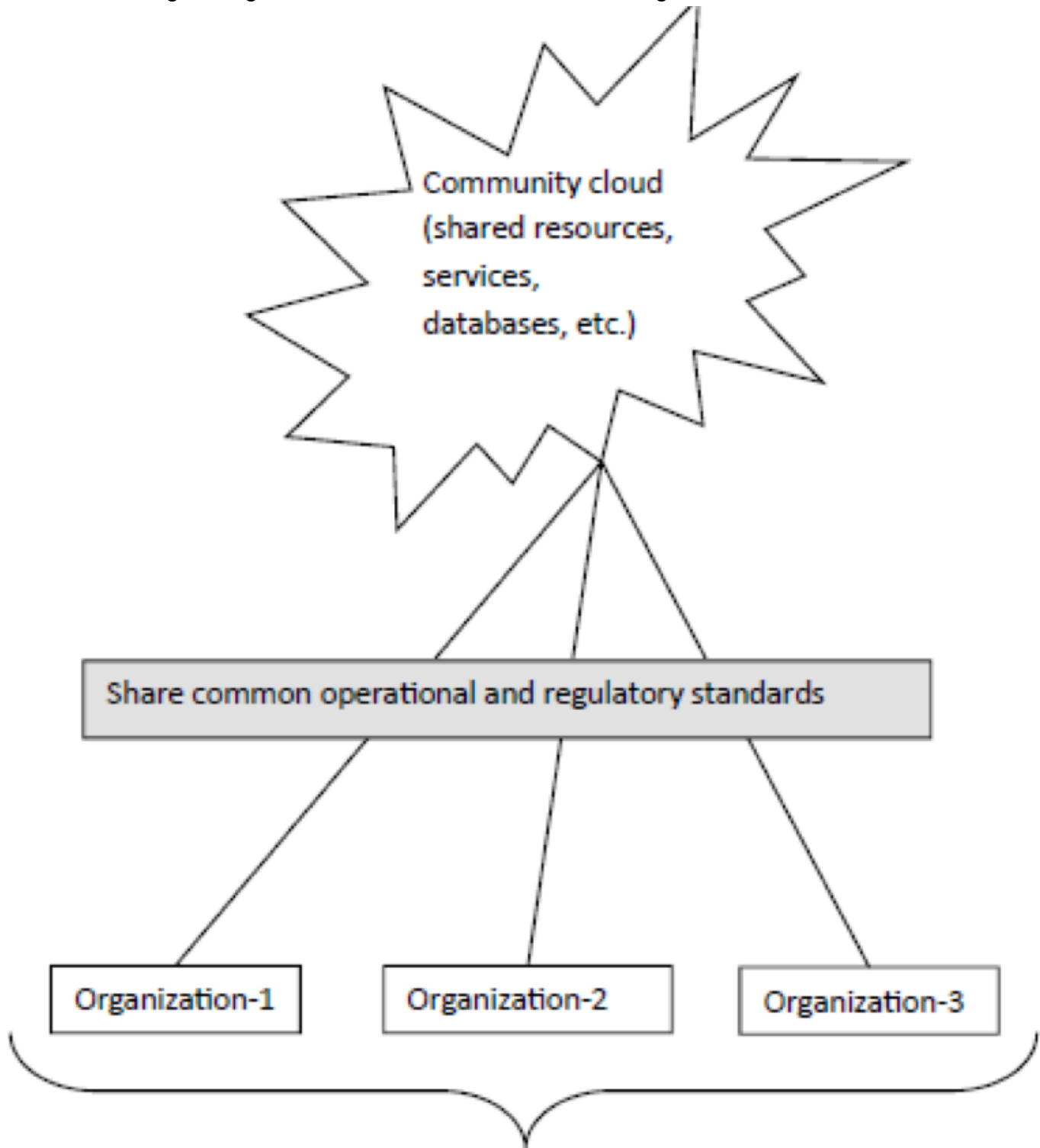
Ci saranno delle operazioni sensibili che possono essere fatte solo attraverso i cloud privati altre invece tramite quello pubblico.

Community Cloud:

un infrastruttura del cloud dove organizzazioni multiple dello stesso dominio condividono risorse e servizi con operazioni comuni e standard di regolazione.

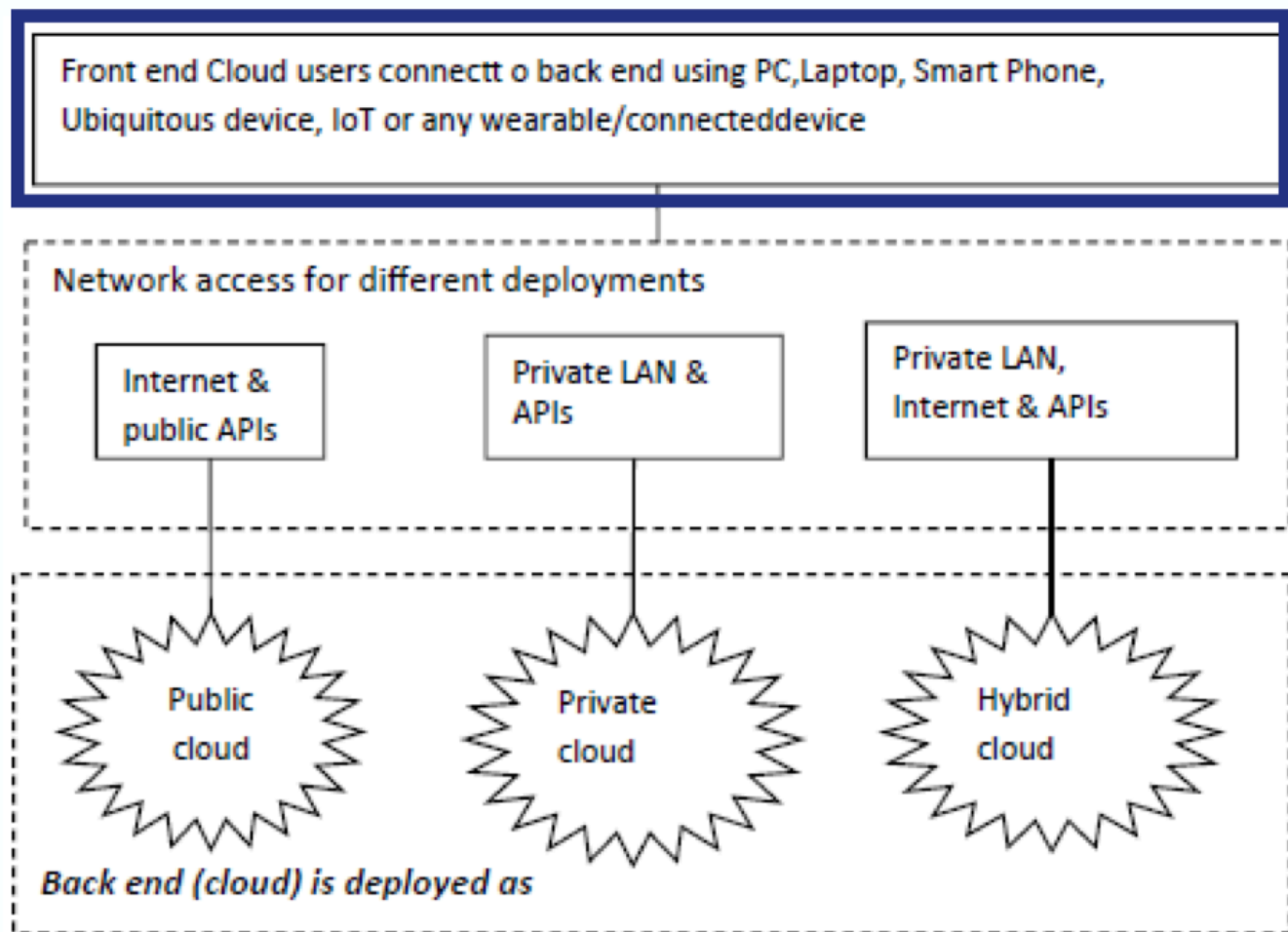
Ci saranno quindi risorse condivise(dati, servizi, risorse), Standard Comuni (policy comune)

Utilizzato in : agenzie governative istituzioni educazionali, organizzazioni sanitarie.



Front End dell'Architettura:

Il Front End è il livello delle API attraverso il quale gli utenti del cloud accedono ai servizi. Gli utenti possono accedere al cloud utilizzando vari dispositivi (cellulare, computer). Quelli che appartengono all'organizzazione possono accedere ai servizi di cloud tramite le loro workstation. Si accede in public tramite Internet mentre in private tramite LAN.



Servizi e Provider

- Principali: Amazon, Salesforce, Microsoft, IBM, Google
- **Esempi AWS:**
 - **EC2:** calcolo on-demand con pricing On-Demand, Reserved, Spot
 - **S3:** storage oggetti altamente scalabile e sicuro
 - **CloudFront:** CDN per ridurre la latenza grazie a distribuzione via edge locations
- **Google App Engine:** PaaS per app web, nasconde l'infrastruttura sottostante

Cloud Networking

È un'infrastruttura cloud dove le risorse di un'organizzazione di rete sono ospitate in un cloud. Il cloud può essere privato, pubblico o ibrido.

Queste risorse sono principalmente:

- software di gestione della rete
- Connettività: facilita l'accesso alle risorse della rete aziendale.

Ci sono diverse tipologie di configurazioni di connettività:

- Connessione Site to Site: connessione sicura tra due o più rete, connette vari uffici con una singola rete aziendale.
- Connessione Point to Site: connette un singolo computer a tutta la rete virtuale e abilita l'accesso remoto per gli impiegati.
- Connettività VNet to VNet: connessione sicura tra due o più virtual networks.
- router virtuali
- firewall
- bilanciatori di carico
- CDN
- VPN
- DNS

Le **Tipologie di rete** sono:

- **Cloud-Enabled Networking:** Il Cloud-enabled Networking si riferisce a un modello in cui l'infrastruttura di rete rimane disponibile presso la sede del consumatore (on-premises). In questo approccio:
 - Gestione della rete: La rete è costruita e la trasmissione dei dati (incluso il routing) è gestita dall'azienda stessa. Ciò significa che l'organizzazione mantiene il controllo diretto e la proprietà della propria infrastruttura di rete fisica.
 - Servizi cloud: I servizi cloud vengono utilizzati per supportare o potenziare l'infrastruttura di rete esistente. Questi servizi includono, ad esempio, il monitoraggio della rete, la gestione della rete e i servizi di sicurezza.
Scopo: L'obiettivo è sfruttare le capacità del cloud per migliorare la visibilità, il controllo e l'efficienza della rete on-premises, senza spostare l'intera infrastruttura nel cloud
- **Cloud-Based Networking:** Il Cloud-based Networking, invece, sposta l'intera infrastruttura di rete o gran parte di essa nel cloud. In questo modello:
 - Gestione del processo di rete: I fornitori di servizi cloud gestiscono l'intero processo di rete, inclusi connettività, controllo, gestione e sicurezza. Questo include la sostituzione delle reti private aziendali tradizionali.
 - Infrastruttura virtuale: I servizi sono forniti attraverso reti virtuali noleggiate (Networking as a Service). Ciò risolve sfide aziendali come la gestione di uffici remoti, firewall, bilanciamento del carico e protezione DDoS, che tradizionalmente richiedevano hardware e configurazioni complesse in sede.

- Nuovo modello di traffico: Questo modello stabilisce un nuovo paradigma di gestione del traffico e può sostituire configurazioni on-premises come le connessioni MPLS.

Benefici: costi ridotti, consegna rapida e sicura, monitoraggio avanzato, visibilità, banda elevata

Trend attuali: SD-WAN per gestione integrata di reti fisiche e virtuali, focalizzazione su reti software-centriche e sicurezza
