

Cloud computing

Monday, 29 May 2023

08:48

- IOT
 - Internet of things
Aka un sistema di oggetti fisici che possono essere scoperti, monitorati e controllati da dispositivi elettronici
 - Things = Oggetto smart
- Cloud computing = Stile di computazione che fornisce un sistema di capacità "servizi"
- Edge computing
 - Sono i dispositivi di confine (telefono, ecc)
 - Nel tempo stanno aumentando di capacità/risorse contenute
 - Questo rende espensivo il trasferimento dei dati
 - C'è un problema di sicurezza/privacy con i dati che si muovono verso il cloud
 - E nota che, non tutti i dati che si trasferiscono non sono essenziali
 - Bisogna comprendere quali dati filtrare
- Sistema smart = Dispositivi + Edge + Cloud + Computer che elaborano
- Cloud computing = Stile di computazione che prevede un servizio scalabile ed elastico che viene dato utilizzando tecnologie informatiche attraverso l'internet
 - API
 - Gli utenti non devono sapere di come vengono controllati, loro utilizzano tecnologia senza sapere come è fatta
 - E' una esperienza utente ed un modello di business
 - L'utente deve essere comodo
 - Ed è un modello di business dove si può guadagnare
 - E' una infrastruttura manageriale metodologica

Un altro significato: E' un modello per attivare, conveniente, quando ti serve con accesso attraverso la rete che ha un pool di risorse

Caratteristiche:

- On demand self-service
Lo possiamo usare quando ci serve ogni volta
- Broad network access
Possiamo accedere attraverso la rete
- Resource pooling
Io ho bisogno di avere un pool di risorse, dove prendere ciò che mi

io ho bisogno di avere un pool di risorse, devo prendere ciò che mi serve

- Rapid elasticity

Se ho 1 cliente ne apro 1, se ho 5 clienti ne apro 3

Devo poter assegnare/disassegnare le risorse a seconda delle situazioni

- Measured service

Devo poter misurare cosa c'è

Ma perchè? Siccome ti devo fare pagare

Esistono 3 modelli di servizio:

- Software as service SAAS

Permette di accedere alla piattaforma

- Platform as service PAAS

Permettere di accedere ed usare l'infrastruttura

- Infrastructure as service IAAS

Voglio poter accedere a dell'hardware a cui posso accedere attraverso un api

[Ci scommetto quanto volete che ci sarà una domanda su questo]

I modi per gestire il cloud computing:

- Pubblico

E possono essere:

- Dedicati
- Community

- Private

E' un modello standard (es. posta elettronica email)

- Ibrido

Parti private e parti pubbliche

- La tecnologia che permette il tutto è la virtualizzazione

- Creare un programma che fa finta di interfacciarsi ad A ma si interfaccia a B

- Abbiamo un man in the middle

- Per fare la virtualizzazione si può mascherare:

- Hardware
- Operazioni di sistema, fornisce interfaccia all'hardware
- System calls, fornisce interfaccia al sistema operativo
- API, interfaccia al sistema operativo implementato con chiamata di funzioni

- Esistono 2 tipologie di virtualizzazioni:

- Processo

- Fornisce un set di istruzioni macchina astratte, e poi i programmi sono compilati in questo linguaggio macchina

- Monitor

- Capacità di fornire una macchina a differenti programmi contemporaneamente, come se ci fossero multiple cpu

- Microservizi

- Implemento ogni funzionalità in un servizio autonomo
- Promuove SE (software engineering) best principi/practice
 - Low coupling, high cohesion
 - Asincrono > sincroni
 - Coreografico > orcheografico
- Però:
 - Si aggiunge complessità
 - Non sempre una soluzione va sempre bene
- Principi:
 - Bisogna costruire servizi che siano leggeri
 - Sono disegnati al principio che ogni modulo/classe dovrebbe avere la responsabilità su singole parti funzionali del software
 - Il microservizio è una cultura, non tecnica
 - Organizzato intorno alle capacità business
 - Prodotto non progetto
 - Un team sviluppo prende controllo e deve essere coordinato con gli altri
 - L'infrastruttura automatica: DevOps: sistema automatici
 - Se io ho N componenti ognuna autonoma, posso decidere come fare
 - ◆ Ognuno può evolvere in modo autonomo
 - ◆ Posso cambiare 1 componente senza modificarli tutti siccome abbiamo una interfaccia
 - L'idea è quella di suddividere l'applicazioni in una collezione di piccoli servizi interconnessi, ed ogni servizio ha una collezione di singole e distinte funzionalità
 - Comunicazione affidata ad un API gateway
 - Ogni servizio ha un suo database e un suo schema
 - Regola Martin Flower
 - Non usarli se hai un sistema semplice
 - La maggior parte dei sistemi non dovrebbero usare questo
 - Non suddividere in microservizi senza la necessità di farlo

