

# Sistemi Informativi:

## Cloud Computing

# Sommario

- Software As A Service (SAAS)
- Servizi forniti su Internet
- HW e SW in Data Centers
- Utility computing
- Virtual Desktop (vedi slide VDI..)

# Aspetti nuovi

- Illusione di infinita potenza computazionale
- Eliminazione di grosso impegno iniziale da parte del cliente
- Possibilita' di pagare potenza computazionale a breve termine in base alle esigenze

# origine

- costruzione di enormi Data Centers
- riduzione di 5-7 volte sui costi di HW, elettricità', costi fissi..
- sfruttamento ottimale servers (statistical multiplexing)
- prezzi inferiori ad un medio Data Center privato, tuttavia redditizi

# Livelli di astrazione

- Amazon EC2 (IaaS)
- Microsoft Azure (PaaS)
- Google AppEngine (SaaS)

# Infrastructure as a Service

- Macchina virtuale su vari OS:  
Linux, Windows, Unix.
  - simile a VirtualBox
- Replicazione
- Scalabilita'
- Monitoring

# IaaS: Amazon EC2

- macchina virtuale HW
- una interfaccia a basso livello per configurare l'HW virtualizzato (pensate agli emulatori).
- nessun limite a priori sul tipo di applicazione, disponendo del controllo su tutte le primitive a basso livello
- difficile supportare scalabilità, guasti/recovery, poiché dipendenti dalla singola applicazione

# Platform as a Service

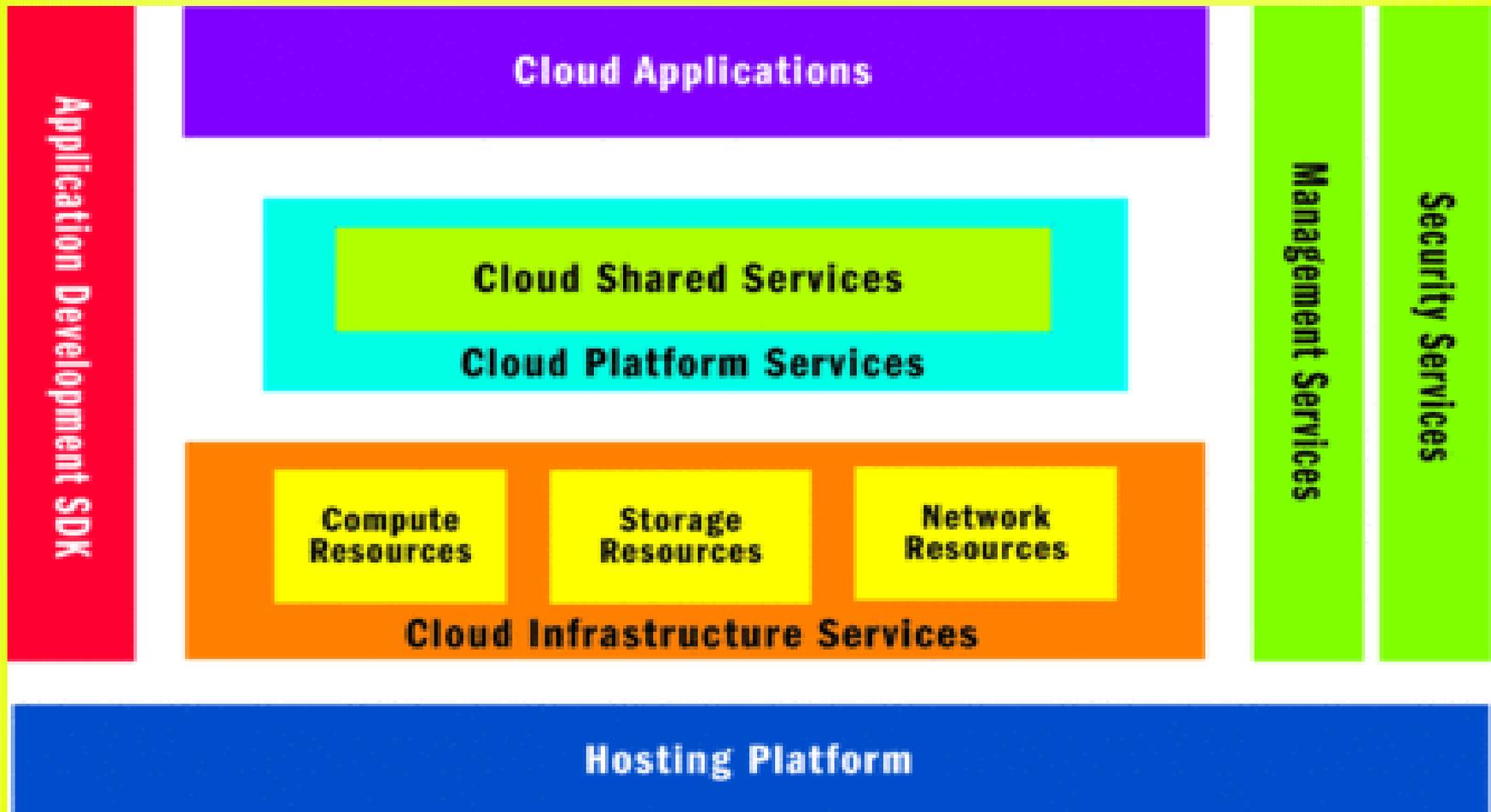
- IaaS +
- Tools di sviluppo
- Accesso a dati e DB

# PaaS: Google AppEngine

- Indirizzato alle applicazioni Web
- Separazione fra logica applicativa e dataStorage
- Il modello request/response limita la CPU dedicata ad ogni singola richiesta
- Scalabilità automatica ed alta disponibilità
- Inadatto ad applicazioni generiche

# Software as a Service

- PaaS +
- Software sul provider
- Connettersi ed usarlo
- Es: social, mail, google apps



# Quando conviene?

- Domanda con picchi variabili nel tempo
- Impossibile prevedere la domanda
- Commutativita' costo/tempo per grossi lavori "batch" (es. analisi statistiche).
- Dati Numero Ore Cloud(NC), CostoOrario(O), Costo Data Center(CD)
- ci interessa il Min ( $NC * O$  , CD)

# Requisiti importanti

- SW “elastico”, scalabile in alto e in basso
- Licenze “pay per use”
- SW di infrastruttura con contabilità preinserita
- HW in unità maggiori, con consumo di energia proporzionale all'uso
- Interfacce standardizzate stimolano l'interoperatività

# Parallelo storico

- Una volta, fabbricazione chip in proprio
- costi raddoppiano ogni generazione
- oggi \$3bn
- Intel e Samsung tuttora
- altri delegano (es NVIDIA) a fabbriche (es Taiwan Semiconductor Manufacturing Company (TSMC). evitando i rischi.

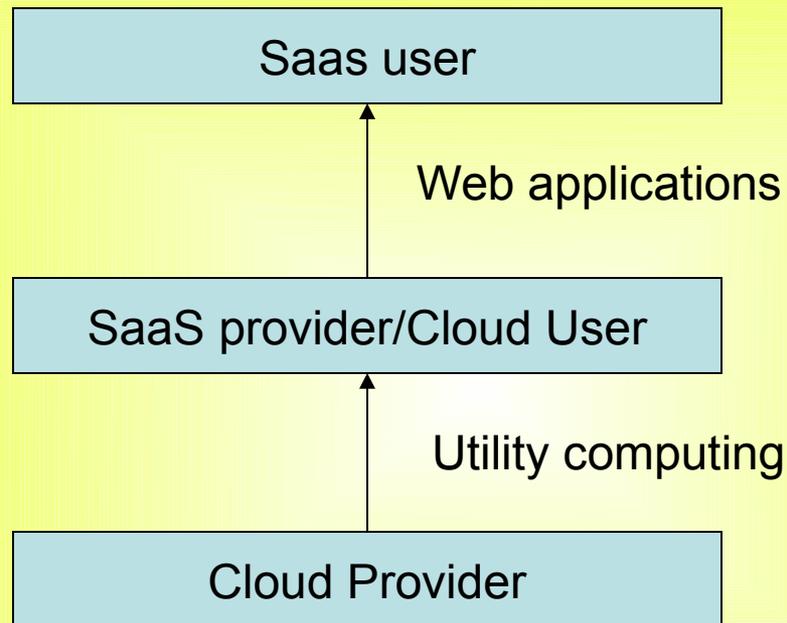
# Critiche rivolte alla definizione

- Sembra includere tutto
- mutiple definizioni
- termine nuovo per vecchie attivita'
- “marketing hype”

# Cloud Computing

- Cloud Computing = SaaS+ Utility Computing
- SaaS per il fornitore, semplifica installazione, manutenzione, controllo versioni
- SaaS per il cliente, sicurezza dati, collaborazione facilitata, “anytime, anywhere”
- Utility Computing per il fornitore permette di non avere il Data Center

# Cloud Computing



L'ultimo livello e' ricorsivo: Un SaaSUser puo' diventare un SaaS provider combinando servizi di altri SaaS providers

# Perche' ora?

- nel 2001 IntelComputing service richiedeva un contratto, non un semplice pagamento orario.
- Oggi Amazon Ec2 fa pagare 1 ora di CPU \$0,1 Amazon S3 fa pagare 1 giga di disco \$0,12 al mese, piu' circa altrettanto per upload o download
- Amazon confida che i suoi utenti non usino tutti contemporaneamente i suoi servizi
- Creare un tale Data Center costa \$100m.

# Enormi Data Centers

- inizio millennio: crescita enorme dei WebServices.
- Amazon, Ebay, Google, Microsoft stavano già costruendoli
- Infrastruttura SW: MapReduce, Google File System, BigTable, Dynamo
- Competenza operativa riguardo sicurezza fisica ed elettronica

# Fornire Clouds?

- Profitto tramite economia di scala: acquisto HW a 1/3 1/7 del costo, SW ammortizzato su tantissime macchine
- Sfruttare investimento esistente, a ridotti costi aggiuntivi (Amazon)
- Conservare la clientela, offrendo una migrazione facile alla nuova piattaforma (Azure).
- Diventare una piattaforma applicativa

# Costi

Technology	Medium-sized DC	Very Large DC	Ratio
Network	\$95 per Mbit/sec/month	\$13 per Mbit/sec/month	7.1
Storage	\$2.20 per GByte / month	\$0,40 per GByte / month	5.7
Administration	~ 140 Servers / Administrator	>1000 Servers / Administrator	7.1

Costi operativi uguali spesso a 1/3 del totale, si sceglie la locazione dove manodopera , tasse , elettricità costano meno

# Nuovo modello

- spostamento verso servizi leggeri, senza contratto, a basso margine, scarso supporto, per individui o piccoli clienti
- PayPal vs VISA
- Google AdSense
- Amazon WS: basta un carta di credito

# Tipi di cloud

- Public: data center virtuale fuori dal firewall
  - Servizio su internet pubblica
- Private: data center dentro al firewall o spazio dedicato dal provider alla tua compagnia
- Hybrid: un mix delle due

# Quali applicazioni su cloud?

- Mobili interattive : spesso incrociano grandi insiemi di dati, alta disponibilita'
- Elaborazione “batch” parallela di molti dati;
  - occorre considerare costo/tempo di spostamento dati
- Analisi dei dati
- Applicazioni scientifiche CPU intensive (matlab)
- Applicazioni grafiche (rendering)

# Economia: trasferimento del rischio

- Anche se le ore di Cloud costano piu' di un server analogo, eliminiamo il rischio di sotto/sovradimensionare l' HW
- tipici servers di un Data Center sono utilizzati tra 5 e 20%
- pochi sovradimensionano deliberatamente
  - spreco
- se sottodimensionano:
  - perdo potenziali clienti
- Animoto: inizio: 50, dopo 3 giorni 3500, poi 200 stabili

# Economia

- se crisi economica, non perdo il deprezzamento dei servers
- posso potenzialmente approfittare della continua caduta dei prezzi HW
- Anche se le apps usano diverse quantità di CPU disco e rete, in casa devo dimensionare per il massimo

# Economia

- \$1 di CPU su un mio server costa \$2.56 su Amazon WS, 10GB \$1.50
- Costi operativi raddoppiano il costo da \$1 a \$2 per CPU
- Amazon crea tre copie dei dati, portando il costo equivalente a \$3

# Esempio

- Suppose a biology lab creates 500 GB of new data for every wet lab experiment. A computer the speed of one EC2 instance takes 2 hours per GB to process the new data. The lab has the equivalent 20 instances locally, so the time to evaluate the experiment is  $500 \times 2 \div 20$  or 50 hours. They could process it in a single hour on 1000 instances at AWS. The cost to process one experiment would be just  $1000 \times \$0.10$  or \$100 in computation and another  $500 \times 0.10$  or \$50 in network transfer fees. So far, so good. They measure the transfer rate from the lab to AWS at 20 Mbits/second. [19] The transfer time is (.....) or more than 55 hours. Thus, it takes 50 hours locally vs.  $55 + 1$  or 56 hours on AWS, so they don't move to the cloud.

# Fattori critici

- Disponibilita' del servizio
- Dati "bloccati"
- Privatezza e verificabilita' dei dati
- Rallentamenti nel trasferimento dati
- Imprevedibilita' delle prestazioni
- Scalabilita' dello storage
- Errori nel software
- Rapida scalabilita'
- Mantenimento della reputazione della cloud
- Pagamento licenze SW

# Disponibilita' del servizio

Service and Outage	Durati on (hrs)	Date
S3 outage: authentication service overload leading to unavailability	2	2/15/08
S3 outage: Single bit error leading to gossip protocol blowup.	8	7/20/08
AppEngine partial outage: programming error	5	6/17/08
Gmail: site unavailable due to outage in contacts system	1.5	8/11/08

# Disponibilita' del servizio

- Poche infrastrutture cosi' affidabili
- per il singolo ente, usare diversi cloud providers
- puo' proteggere per l'attacco "Denial of Service", scalando velocemente e dando tempo per reagire, oltre a costare molto all' attaccante

# Dati “bloccati”

- poca standardizzazione API, vista la novità dei servizi
- pericolo che il fornitore possa fallire, che i prezzi aumentino
- Se API standard, usare diversi Cloud Providers
  - paura (per i providers) che i prezzi crollino

# Privatezza e verificabilita' dei dati

- Crittografare prima di memorizzarli
- problemi legislativi di luogo fisico di memorizzazione (dentro la nazione)

# Rallentamenti nel trasferimento dati

Come fare?

spedire dischi fisici!

affidabile

costa meno ed e' piu rapido in certe condizioni

una volta su cloud, I dati potrebbero attrarre altre applicazioni

# Rallentamenti nel trasferimento dati

To quantify the argument, assume that we want to ship 10 TB from U.C. Berkeley to Amazon in Seattle, Washington. Garfinkel measured bandwidth to S3 from three sites and found an average write bandwidth of 5 to 18 Mbits/second. [19] Suppose we get 20 Mbit/sec over a WAN link. It would take (.....)more than 45 days. Amazon would also charge you \$1000 in network transfer fees when it received the data. If we instead sent ten 1 TB disks via overnight shipping, it would take less than a day to transfer 10 TB and the cost would be roughly \$400, an effective bandwidth of about 1500 Mbit/sec.<sup>11</sup> Thus, “Netflix for Cloud Computing” could halve costs of bulk transfers into the cloud but more importantly reduce latency by a factor of 45.

- **Imprevedibilita' delle prestazioni**
- il problema e' correlato all' IO su disco; le VM potrebbero scrivere su FlashMemory per accelerare questa operazione
- **Scalabilita' dello storage**
- tuttora argomento di ricerca
- **Errori nel software**
- migliori debuggers!
- **Rapida scalabilita'**
- tecniche statistiche di machine learning
- **Conservazione della reputazione:**
- Creare servizi di certificazione

# Pagamento licenze SW

- Ora periodico per il SW commerciale
- I cloud providers hanno così usato molto Open Source SW
- I commerciali devono cambiare stile di vendita:
- “10000 ore di tal CPU presso Amazon”

# In futuro ..

- Quali livelli di infrastrutture Cloud prevarranno? a basso, ad alto livello oppure una varietà?
- ci sarà una standardizzazione delle API, che potrebbe potenzialmente portare ad una guerra dei prezzi?
- I fornitori di servizi saranno indipendenti o verranno inglobati dai cloud providers?