

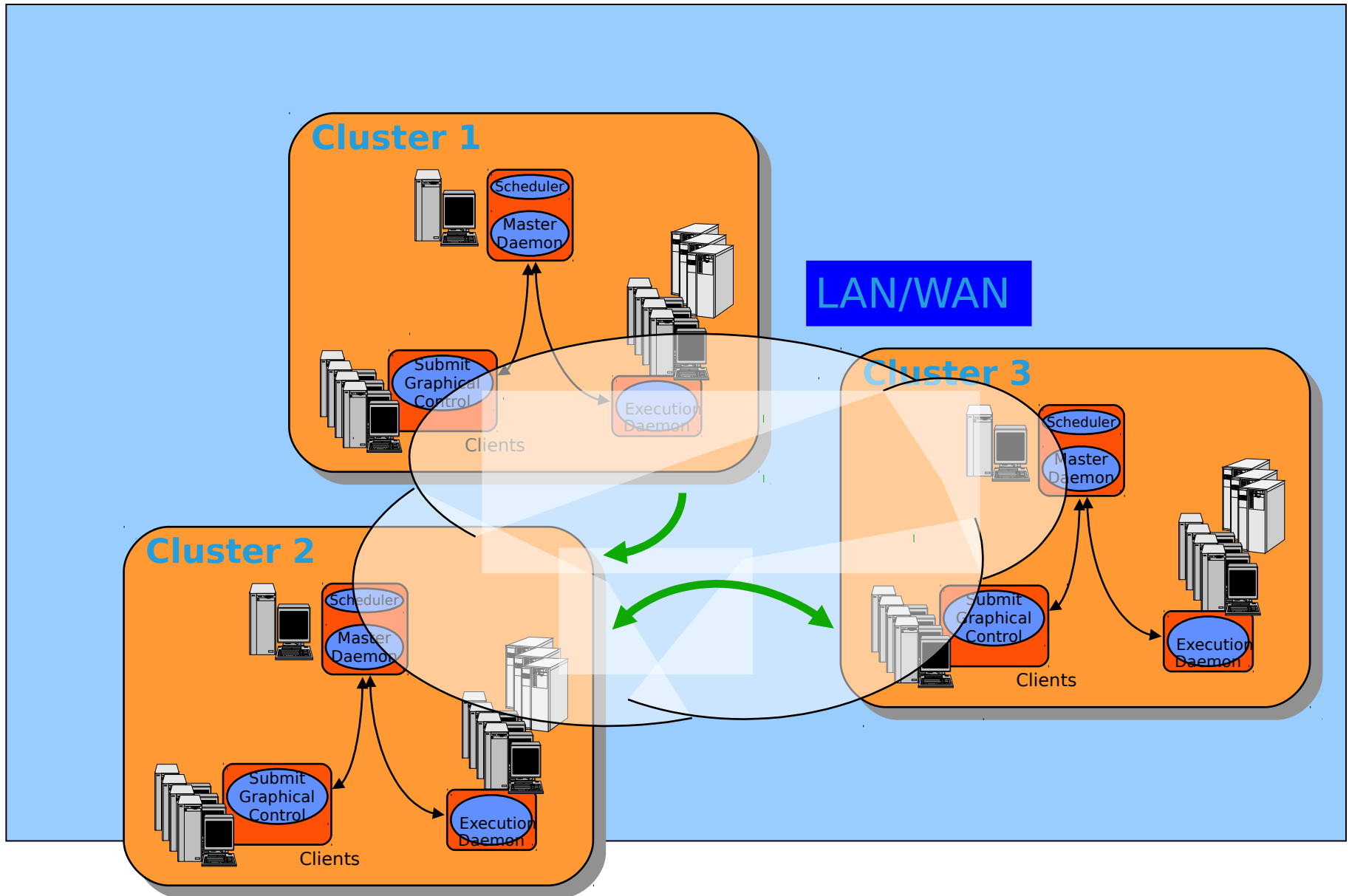
# **Evoluzione delle risorse di calcolo: Grid, Cloud e Virtualizzazione**

Introduzione al Grid Computing

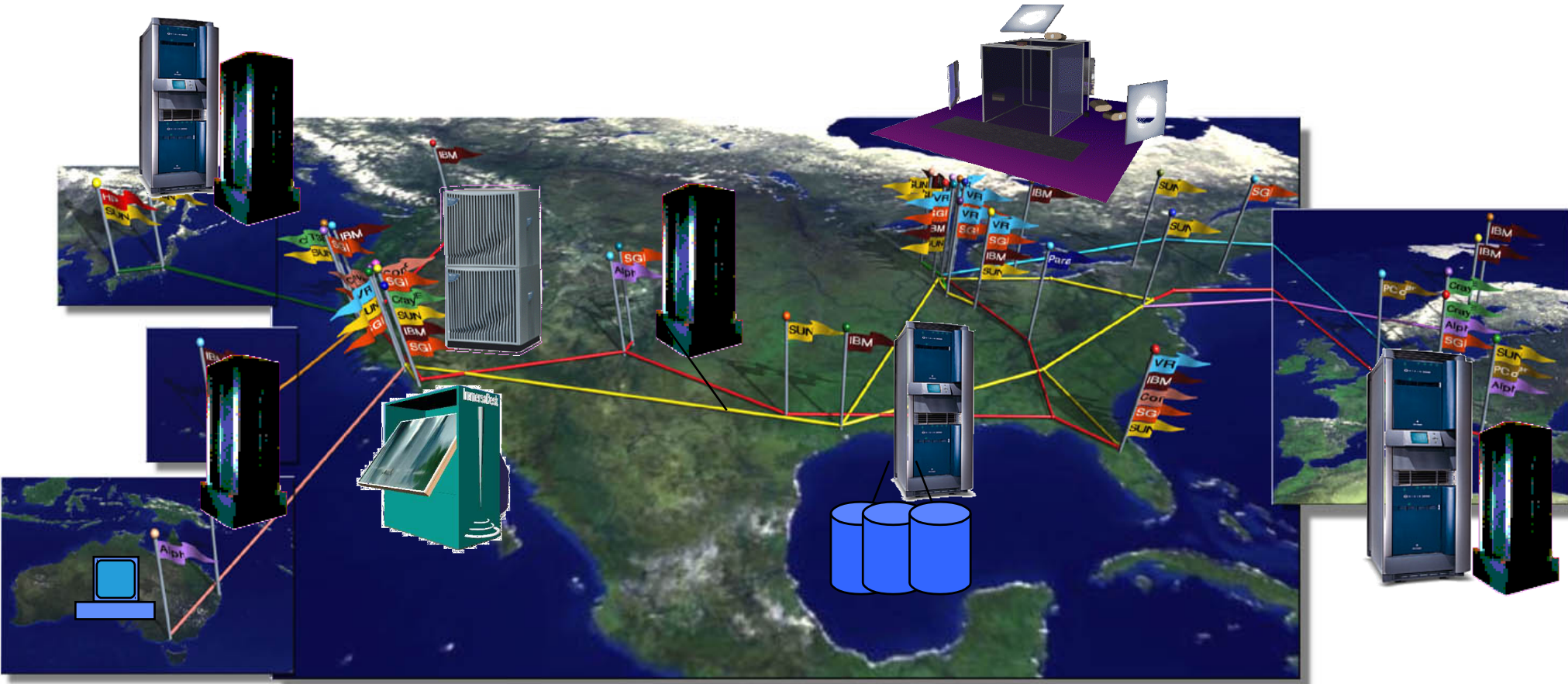
Introduzione al Cloud Computing

La virtualizzazione nel Cloud Computing

# Clusters of Clusters (HyperClusters)



# Towards Grid Computing....



# Grid Computing

**La Grid è nata per connettere supercomputers.**

**Per avere sistemi HPC evitando di costruire di nuovi supercomputers.**

**Le applicazioni iniziali erano solo scientifiche (high energy physics, simulazioni militari, astrophysics, ...)**

**Ma anche il Web è nato per applicazioni scientifiche!**

# The Grid Today

**Dopo dieci anni le applicazioni di Grid ormai sono moltissime.**

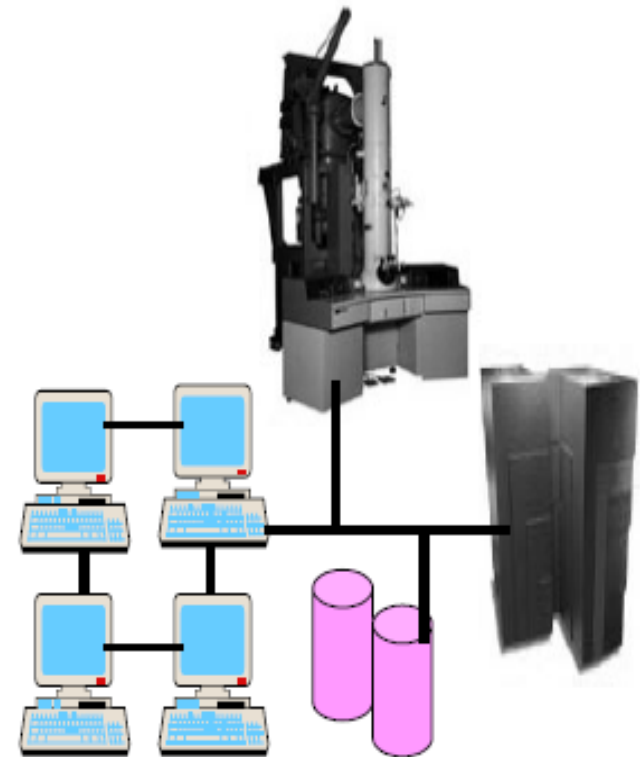
**Dalle applicazioni scientifiche si è passati alle applicazioni industriali e di business.**

**Alcune delle aziende che usano questa tecnologia sono: SAP, Oracle, Sun, IBM, Ferrari, HP, Intel, BBC, British Telecom, TelefonicaTelefonica, Audi, , .....**

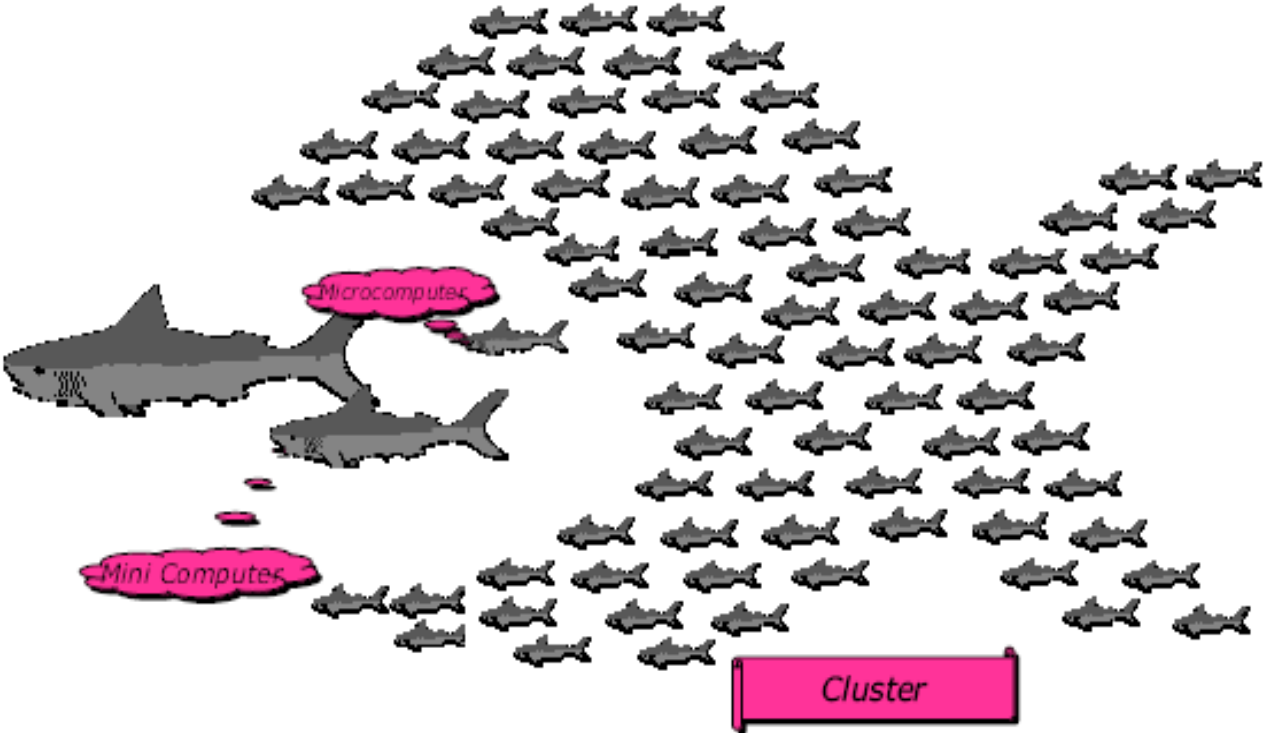
# What is Grid Computing?

---

- **Computational Grid**
  - A collection of distributed, possibly heterogeneous resources which can be *used as an ensemble* to execute large-scale applications
  - *Metacomputer*
- **Term Computational grid comes from an analogy with the electric power grid:**
  - Electric power is ubiquitous
  - Don't need to know the source (transformer, generator) of the power, or the power company that serves it
  - This analogy regards the search for machine cycles
- **Other specialized Grids are possible**
  - **Data Grid**, i.e. a Grid used for searching, accessing and elaborating distributed data

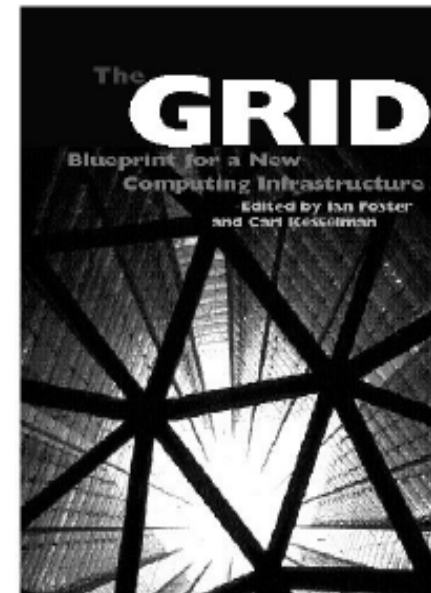
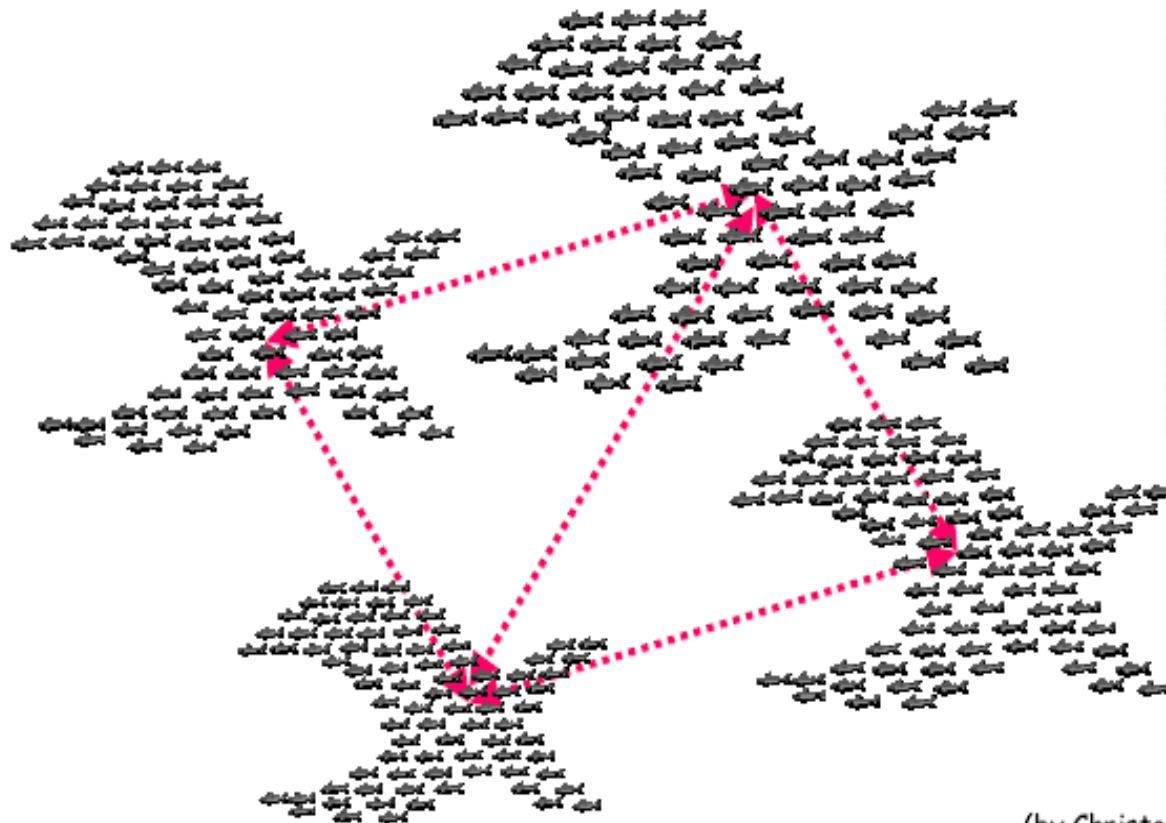


Yesterday ...



(by Christophe Jacquet)

...and today



see also [www.grid2002.org](http://www.grid2002.org)

(by Christophe Jacquet)



# What is Grid Computing?

---

- Coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations (Vos)

[ I.Foster]

- **A VO is a collection of users sharing similar needs and requirements in their access to processing, data and distributed resources and pursuing similar goals**

- **Key concept :**

- ability to negotiate resource-sharing arrangements among a set of participating parties (providers and consumers) and then to use the resulting resource pool for some purpose

[I.Foster]

# What is the Grid ? (1)

"Like its namesake, a Grid , is a mix of technology technology, infrastructure, and, standards".

**The technology** is software that allows resource providers (whether individuals or institutions institutions) to federate computers, storage, data, networks, and , other resources and for resource consumers to harness those federated resources when needed.

**The infrastructure** comprises the physical hardware and services that must be maintained and operated for this resource federation and access to occur.

**Standards** codify the messages that must be exchanged, and the policies that must be followed, to achieve those goals.

# Grid Infrastructure

## An infrastructure that couples

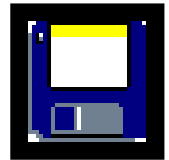
Computers (PCs, workstations, clusters, traditional supercomputers, and even laptops, notebooks, mobile computers, PDA, etc)

Databases (e.g., transparent access to human genome database)

Special Instruments (e.g., radio telescope--SETI@Home Searching for Life in galaxy, Austrophysics@Swinburne for pulsars)

People (may be even animals who knows, frogs already planned?)

across the local/wide-area networks (enterprise, organisations, or Internet) and presents them as an unified integrated (single) resource.



## What is the Grid ? (2)

"Grid computing uses open standards and protocols to enable the the virtualisation of distributed computing resources to create a single system image across eterogeneous, dynamic, geographically dispersed IT environments."

# Grid come Internet

“Una Grid” è una infrastruttura di comunicazione e condivisione di risorse di elaborazione eterogenee e dinamiche.

N.B.: Ce ne sono tantissime

“The Grid” è:

... like "the Internet", denotes the global set of computers that speak the same protocols. In that sense, "the Grid" , is a work in progress, as relevant standards continue to be codified and adopted.

# Many GRID Projects and Initiatives

## **PUBLIC FORUMS**

Computing Portals  
Grid Forum  
European Grid Forum  
IEEE TFCC!  
GRID'2000 and more.

## **Australia**

Nimrod/G  
EcoGrid and GRACE  
DISCWorld

## **Europe**

UNICORE  
MOL  
METHODIS  
Globe  
Poznan Metacomputing  
CERN Data Grid  
MetaMPI  
DAS  
JaWS  
and many more...

## **Public Grid Initiatives**

Distributed.net  
SETI@Home  
Compute Power Grid

## **USA**

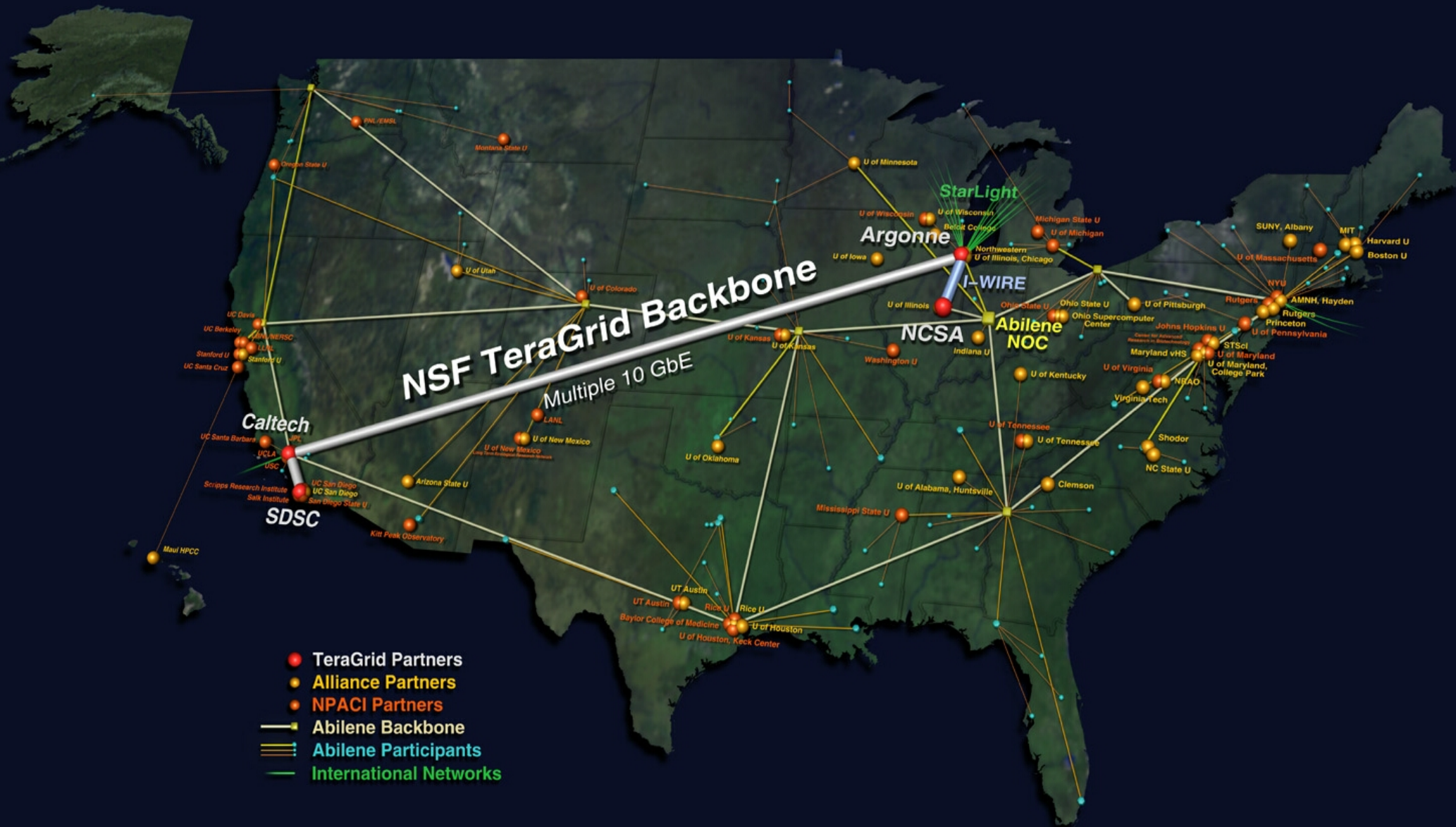
Globus  
Legion  
JAVELIN  
AppLes  
NASA IPG  
Condor  
Harness  
NetSolve  
NCSA Workbench  
WebFlow  
EveryWhere  
and many more...

## **Japan**

Ninf  
Bricks  
and many more...

<http://www.gridcomputing.com/>

# Current Supercomputing Grid

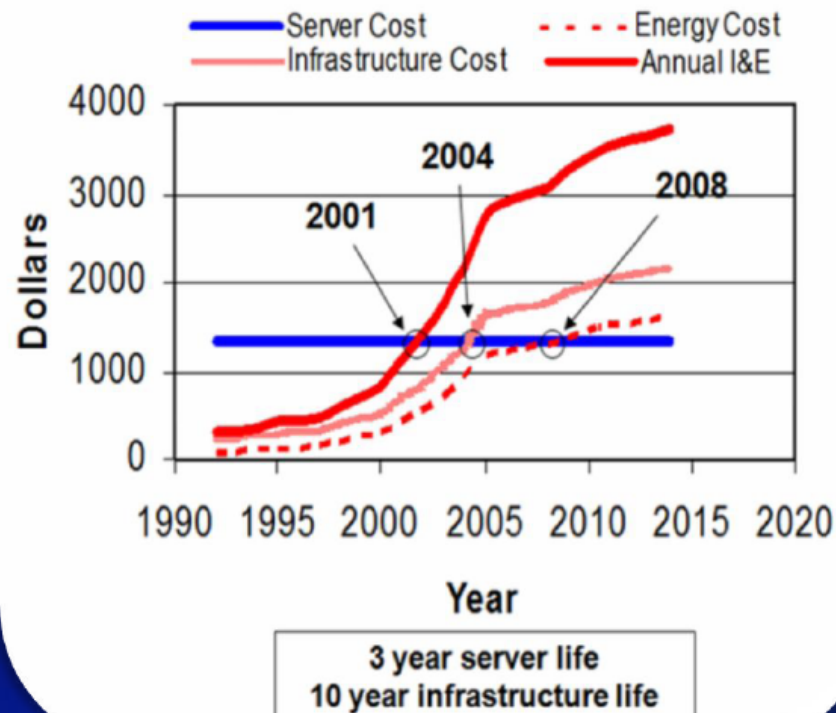


# **SYSTEM TRENDS**



# The New Scaling Economics

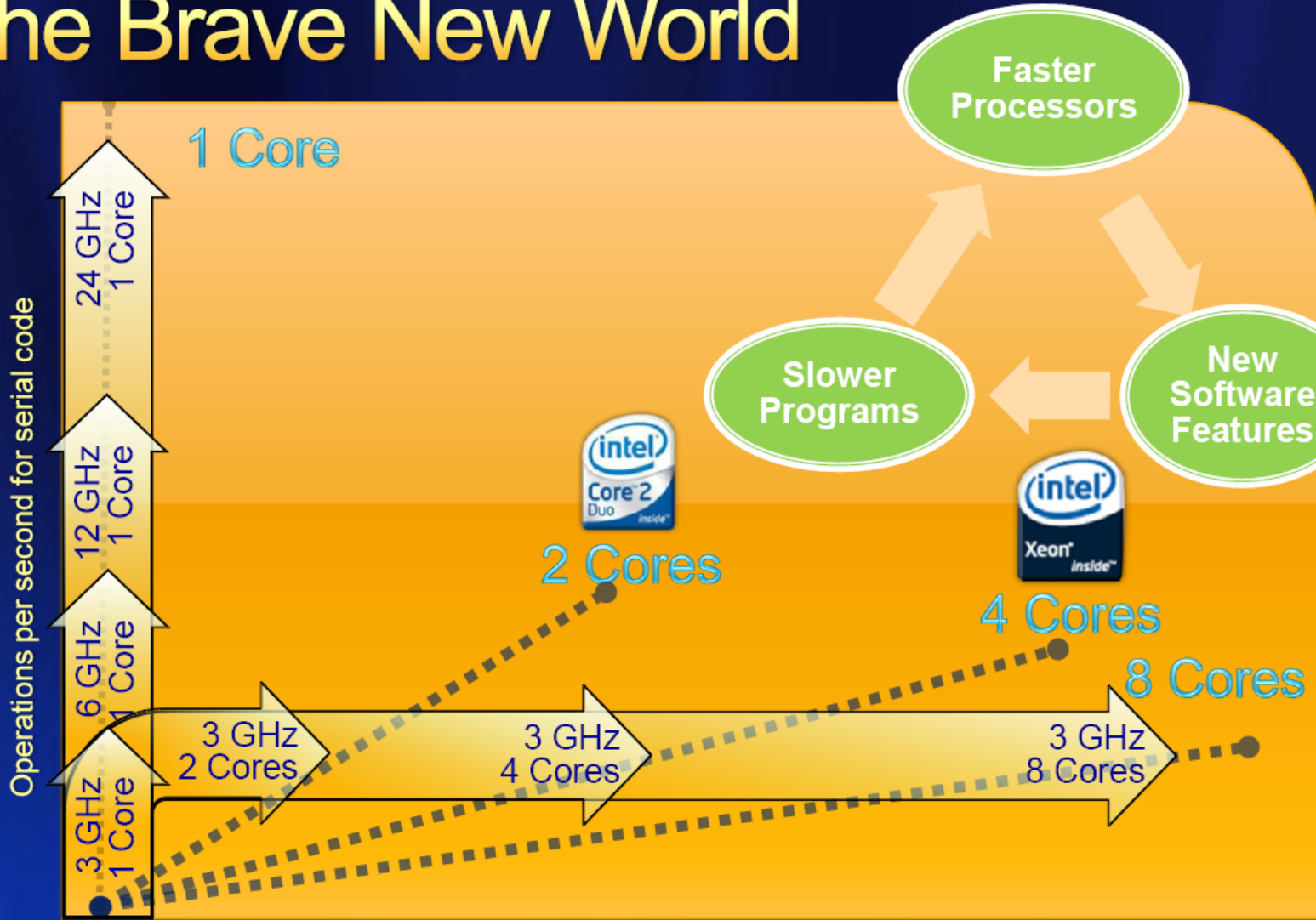
Annual Amortized Costs in the Data Center for a 1U Server



Belady, C., "In the Data Center, Power and Cooling Costs More than IT Equipment it Supports", *Electronics Cooling Magazine* (February 2007)

# The Brave New World

Free Lunch For Traditional Software



Additional operations per second if code can take advantage of concurrency

No Free Lunch For Traditional Software

# Why the World Goes Multi-core I

- Let **f** be the frequency of the processor and **ipc** the actual or sustained **i**nstructions **p**er **c**lock-tick

then we get:      **Perf** = **f** \* **ipc**

- To get the frequency **f** up we just need to increase the voltage **V** (and make the chip smaller),

because:      **f**  $\approx$  **V** (in region of interest)

# Why the World Goes Multi-core II

- **But now the bad news:** the power **E** burned by the chip depends on capacity **C** (transistors etc.), **V** and **f**

$$\mathbf{E} = \mathbf{C} * \mathbf{V}^2 * \mathbf{f} \quad \text{but because } \mathbf{f} \approx \mathbf{V}$$

$$\mathbf{E} = \mathbf{C} * \mathbf{V}^3$$

- So if we reduce the voltage **V** by 20% we still get 0.8 of the original performance. But **E** goes down dramatically because  $0.8 * 0.8 * 0.8 \approx 0.5!$
- Relative performance  $0.8 \times 2 = \mathbf{1.6}$

# Some interesting data

Eight years ago: ASCI White (IBM)

- #1 on the Top500 in June 2001



Peak performance: 12,288 Gflop/s  
Weight: 106 Tons (w/ 160 TB storage)  
Power: 3MW  
Cost: \$110 million

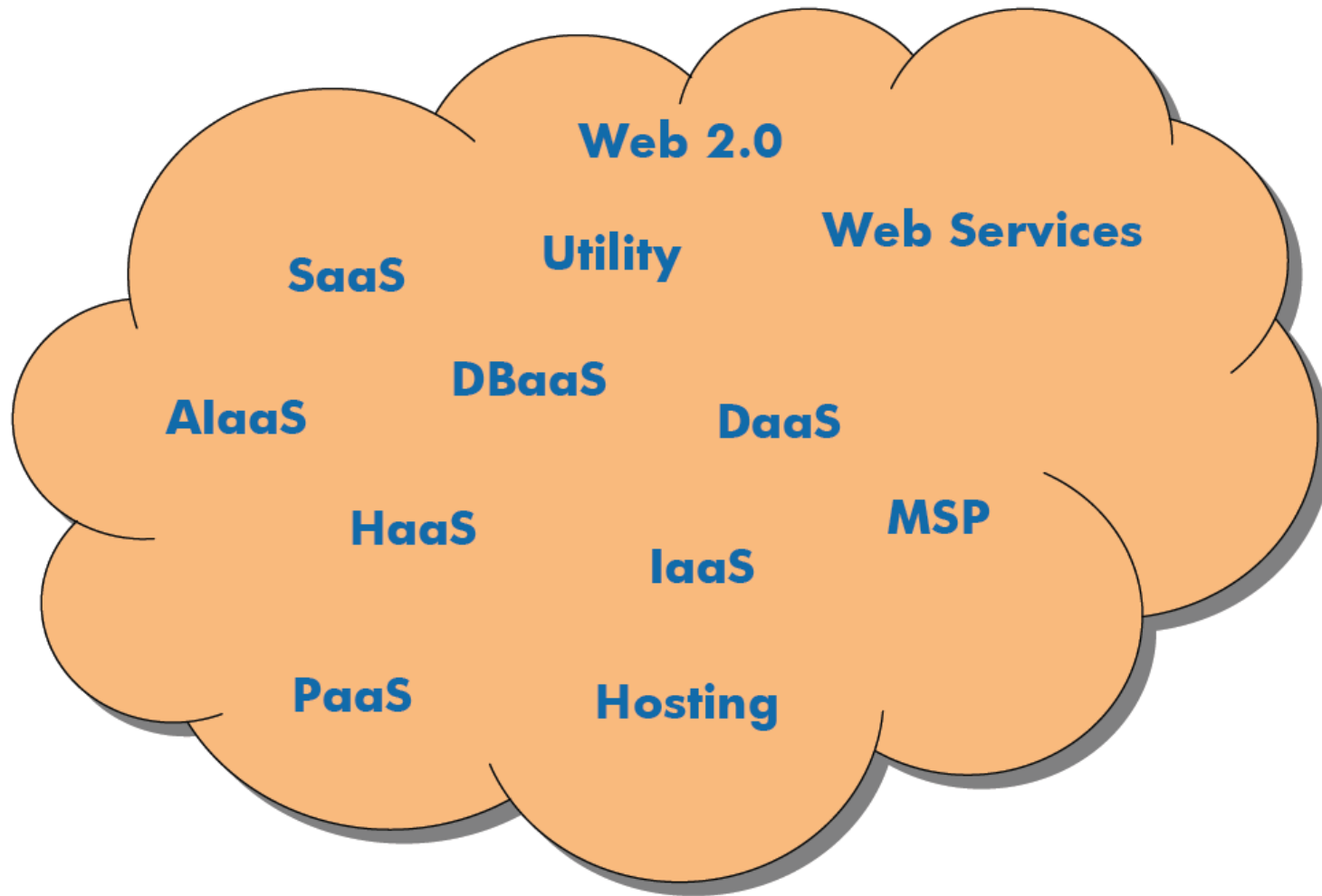
Today: one rack full of B12x220c



Peak performance: 12,288 Gflop/s  
Weight: ~2000 lbs (~1 ton) – 100x lighter  
Power: ~27KW – 100x less power  
Cost: ~ \$800K – more than 100x lower cost

# **SOFTWARE TRENDS**

# The Cloud



# Definizione di Cloud Computing

Il Cloud Computing è un modello comodo per attivare in maniera **on-demand**, attraverso la rete, risorse computazionali configurabili, che possono essere rapidamente richieste e rilasciate con il minimo sforzo ed in maniera self-service. Questo modello promuove affidabilità ed è composto da

5 essenziali caratteristiche,  
3 modelli di servizio e  
4 modelli di implementazione.



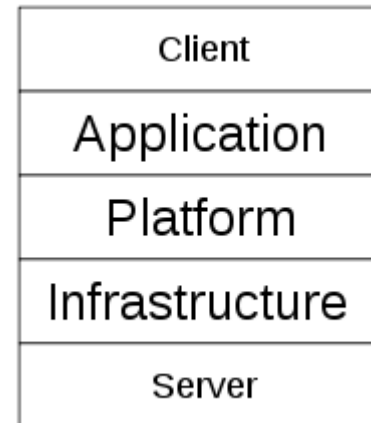
# Definizione di Cloud Computing

I modelli di servizio sono quelli citati (**IaaS, PaaS, SaaS**).

I modelli di implementazione sono noti come **Private Cloud, Public Cloud, Community Cloud e Hybrid Cloud**.

Le caratteristiche essenziali, infine:

- **On-demand self service,**
- **broad network access,**
- **resource pooling,**
- **rapid elasticity,**
- **measured service**



# IaaS PaaS SaaS

il Cloud Computing è visto dagli utenti in modo molto differente a seconda del proprio background informatico. E questa differenziazione combacia con i tre modelli di servizio ormai a tutti noti, (noti quasi sempre solo come sigla), **IaaS (Infrastructure as a Services)**, **PaaS (Platform as a Services)**, **SaaS (Software as a Services)**. Maggiore è la profondità delle conoscenze informatiche e maggiore è l'identificazione del Cloud Computing con il relativo modello di servizio.

**Sistemisti ed "Hardweristi"** sono molto Infrastructure as a Services (es. tutti coloro che hanno vissuto la virtualizzazione). **Sviluppatori e programmatori** sono più Platform as a Services (esempio quelli abituati all'hosting classico). **I commerciali, i comunicatori, gli uomini di marketing, i dirigenti d'azienda**, e la maggior parte di tutte le altre figure che si affacciano ad internet si limitano a vedere il Cloud come Software as a Services.

Il SaaS è l'area di maggior diffusione e sviluppo del Cloud Computing, ma è anche quella dove è più facile trovare dei rebranding di prodotto per rilanciarne le vendite. Le aree che maggiormente introducono innovazione elasticità e vantaggi competitivi sono le altre due IaaS e PaaS.

# Il Cloud Computing

Un sempre maggior numero di applicazioni software di utilizzo corrente sono (o diventeranno) applicazioni Web, quindi di fatto applicazioni SaaS (**Software as a Service**) residenti ed eseguite **su computers di terzi** nonché mantenute da questi ultimi, disponibili praticamente "on the cloud", vale a dire sulla rete Internet (il **Web**) vista come una gigantesca piattaforma operativa integrata: tali **applicazioni** sono o saranno perciò usufruibili **on-demand** e a mò di servizio.

# Il Cloud Computing - 2

Le applicazioni di "**Office on the Cloud**", uno dei paradigmi applicativi che si sono affermati più di recente, e consistente nella **possibilità da parte delle aziende di affidarsi ad applicazioni Web per gestire le proprie attività di business ed i propri dati**, aziende che così possono fare a meno di ricorrere a software acquistati, licenziati, oppure sviluppati e mantenuti "in casa": pensate ai servizi offerti da Salesforce.com, alle Google Apps Premier Edition, ma anche a diversi altri servizi.

# Il Cloud Computing - 3

Inoltre, un ulteriore paradigma emergente è rappresentato dal cloud computing, cioè dalla possibilità di servirsi di "**nuvole di computers**" i quali, lavorando come se fossero un tutt'uno, sono in grado di eseguire complesse applicazioni software distribuite di tutti i generi, delle applicazioni che sono capaci di sostenere dei **carichi elaborativi altamente variabili in maniera dinamica** (sono applicazioni **scalabili**, cioè), in dipendenza dell'uso contemporaneo che di esse viene fatto da parte degli utenti. Il motore di ricerca **Google** rappresenta, ad esempio, una di queste superapplicazioni, ma anche quelli di Yahoo! o di Microsoft, per intenderci.

# Il Cloud Computing e Google

Il Google Search Engine viene appunto eseguito all'interno di una piattaforma/infrastruttura di cloud computing, **un gigantesco sistema parallelo e distribuito fatto di computers low-cost** ed il cui "cuore pulsante" è rappresentato da **MapReduce**, uno strato di software (framework) al di sopra del quale Google fa attualmente "girare" più di **10.000 differenti programmi**: un "ciclotrone" in possesso di Google, in sostanza, così come lo ha definito Eric Schmidt, e **che l'azienda di Mountain View vuole mettere anche a disposizione delle aziende e degli sviluppatori come servizio di cloud computing.**

**Il cloud computing è una delle 10 tecnologie strategiche già nel 2008,**  
e sono/saranno in tanti a competere in questo altissimamente strategico  
settore dell'IT:

Google, con la sua infrastruttura;

Yahoo!, che sulla piattaforma Apache Hadoop ha appena lanciato la Yahoo! Search

Webmap utilizzando un cloud di 10.000 CPUs;

Microsoft (Windows Azure);

IBM (che ha recentemente lanciato Blue Cloud, il progetto RESERVOIR, e che sta  
attualmente lavorando al progetto Kittyhawk);

Amazon (con i suoi già utilizzatissimi servizi EC2 e AWS),

Sun Microsystems (con Network.com);

ed una serie di aziende più piccole - tra le quali spicca 3Tera (con il suo AppLogic), Mosso,  
RightScale. Ma ci sono anche Dell, HP, Oracle, ed EMC.

# Defining the cloud

- Everybody is trying to figure out what it is (remember grid?)
- Gartner: “Cloud computing is a style of computing where massively scalable IT-related capabilities are provided ‘as a service’ across the Internet to multiple external customers”
- Forrester: “A pool of abstracted, highly scalable, and managed infrastructure capable of hosting end-customer applications and billed by consumption”



# Grid and Cloud Today (Courtesy Martin Walker, HPCD)

## GRID

- open standards (OGF ...)
- publicly funded & operated (slow evolution)
- no central management
- interoperability important
- geographically distributed; locally owned and managed
- share (usually modest) local resources
- scientific research, high-end users

## CLOUD

- no standardized interfaces
- privately funded & operated (fast evolution)
- managed by a single entity
- no interoperability
- geographically distributed; centrally owned and managed
- make huge systems available
- enterprise applications, information processing, data mining

# Example: Useful cloud computing at the NY Times

- generate pdf files of 11M public domain NY Times articles from 1851 to 1922 by gluing together multiple TIFF images per article
- programmed with Hadoop
- under 24 hours using 100 instances on Amazon EC2
- Derek Gottfried 1 Nov 2007  
<http://open.blogs.nytimes.com>



## Welcome to Google App Engine

---



### **Run your web applications on Google's infrastructure.**

Google App Engine enables developers to build web applications on the same scalable systems that power our own applications.

### **No assembly required.**

Google App Engine exposes a fully-integrated [development environment](#).

### **It's easy to scale.**

Google App Engine makes it easy to design scalable applications that grow from one to millions of users without infrastructure headaches.

### **It's free to get started.**

Every Google App Engine application will have enough CPU, bandwidth, and storage to serve around 5 million monthly pageviews.

# Cloud Computing Vendors

- <http://www.johnmwillis.com/mysql/cloud-vendors-a-to-z/>
- 3Tera, Adobe Air, Akamai, Amazon EC2, Amazon S3, Amazon SimpleDB, Apache CouchDB, Apache Hadoop, Areti Internet, Box-Net, Cassatt Corporation, Citrix (XenSource), CohesiveFT, Dell DCS, Elastra, EMC Mozy, Enki, Enomoly, Enomoly ElasticDrive, EnterpriseDB, Flexiscale, Fortress ITX, Google Apps, HP AlaaS, IBM Blue Cloud, iCloud, Joyent, JungleDisk, Layered Technology, LongJump, Microsoft SSDS, MorphExchange, Mosso, Rackspace, Rightscale, Salesforce.com, Sun Caroline, Sun MySQL, Terremark, VMWare

# Targeting the Convergence of Massive Scale-Out

**Enterprise/HPC**

- Engineering & Geo-Sciences
- Life & Materials Sciences
- Defense/Security
- Scientific Research

**Web 2.0**

- Internet Commerce
- Interactive Media
- On-Line Gaming



## Massive Scale-Out

- Financial Analytics
- Digital Content Creation
- Streaming Media

### Emerging business models:

Best performance = faster time to market

### New metrics:

Best performance per watt, best performance per sq. ft.

### Extreme pain points:

Data center constraints of power, cooling, space, manageability & automation of dynamic workloads

### Market Requirements

Multiple go to market motions, optimized supply chain & unique products and services

# Il “lato oscuro” del Cloud Computing

C'è qualcuno che però argomenta circa l'esistenza di possibili "lati oscuri" del cloud computing medesimo, così come del SaaS.

Un articolo di Forbes, ad esempio, citando Lee Tien, avvocato nonché privacy analyst di EFF, parla dei **rischi riguardanti la perdita di controllo sui dati sensibili che chiunque si affida a queste nuove tecnologie potrebbe correre mettendo i propri dati "nelle mani di terzi"**.

Quanto ai dati sensibili, invece, l'articolo cita TJX, una società a cui sono stati rubati per via telematica i numeri di quasi 46 milioni di carte di credito, parla del governo britannico (e della perdita dei 2 CD contenenti informazioni sensibili su 25 milioni di suoi contribuenti).

Nell'articolo Forbes parla però inoltre di cifratura o crittografazione dei dati (via PGP o TrueCrypt) quale potenziale rimedio a tutto ciò.

# Cloud Computing e Open Source

Nel caso di servizi online e applicazioni web, in cui il software non viene redistribuito all'utente, le aziende possono alterare e adattare il codice senza rendere pubbliche le modifiche, come solitamente previsto ad esempio dalla General Public License, e senza violarne la licenza.

## **L'obsolescenza delle licenze open source nei confronti del Web 2.0. e del Cloud Computing**

<http://www.webnews.it/news/leggi/9249/stallman-si-scaglia-contro-il-cloud-computing/>

La soluzione a tale problema è arrivata con la **licenza AGPL** (Affero General Public License) la quale prevede appunto che il codice rilasciato sotto tale licenza e poi modificato debba essere a sua volta reso pubblico in ogni caso, anche se il programma gira in remoto, che è proprio il caso delle applicazioni Web 2.0.

<http://www.webnews.it/news/leggi/8829/agpl-quando-loopen-source-si-fa-20/>

viene proposto l'utilizzo di un **paradigma zero-knowledge** che permetta una maggiore tutela della privacy degli utenti. Il paradigma zero-knowledge prevede la cifratura dei dati prima dell'invio al server, rendendo così il provider del servizio incapace di accedervi.

# **Cloud Computing: Una opportunità**

**La prossima Google potrebbe non possedere alcun server**

**Idee?**



# Un esempio Italiano: NICE

## **Quanto costa avviare un'azienda tecnologica da zero basata sull'opensource e sul cloud computing?**

Quanto potrebbe costare avviare from scratch (da zero) un'azienda tecnologica di servizi web 2.0 o 3.0, se essa costruisse la propria infrastruttura informatica basandosi esclusivamente su software **open source** e sul **cloud computing**?

Tenendo conto soprattutto dei possibili costi riguardanti l'hardware, le risorse umane, il supporto al software open source stesso, nonché quei costi riguardanti il potenziale consumo di risorse di calcolo propri dell'utility computing. Perché **acquisire il software open source non costerebbe praticamente nulla** a questa ipotetica azienda, e di codice aperto disponibile per le più svariate esigenze di creazione di servizi web **su Internet c'è ne più che abbastanza.**

# Quanto costa avviare un'azienda tecnologica da zero basata sull'opensource e sul cloud computing? - 2

Ha bisogno di un web server? **Apache (XAMPP)**.

Ha bisogno di un database? **MySQL o PostgreSQL**.

Ha bisogno di un router, firewall, VPN? **Vyatta**.

Ha bisogno di un ambiente/piattaforma di virtualizzazione? **Xen o Open Virtual Machine Tools (VMWare)**.

Ha bisogno di diversi strumenti di sicurezza? **netfilter/iptables, OpenVPN, Nessus, Snort, TrueCrypt, ecc.**

Ha bisogno di un CMS? **Drupal o Joomla!**

Ha bisogno di una piattaforma di blogging? **WordPress**.

Ha bisogno di una piattaforma wiki? **MindTouch Deki Wiki o MediaWiki**.

Ha bisogno di creare applicazioni web avanzate? **OpenLaszlo**.

Ha bisogno di creare un servizio di social bookmarking? **Scuttle o SemanticScuttle**.

Ha bisogno di creare un servizio di social networking? **Elgg o PHPizabi**.

Ha bisogno di creare un servizio di diffusione delle news? **NewsCloud**.

Ha bisogno di creare un servizio di video sharing? **Plone/Plumi**.

Ha bisogno di creare una piattaforma Digg-like? **Pligg**.

Ha bisogno di creare un suo servizio di web analytics? **Piwik**.

Ha bisogno di creare ambienti integrati di collaborazione e/o di conferencing? **Openfire, ICEcore, VMukti, Zimbra**.

Ha bisogno di creare applicazioni aziendali e/o di CRM anche in qualità di servizi web? **GNU Enterprise**.

Ha bisogno di creare un knowledge exchange network? **Spree**.

Ha bisogno di creare un motore di ricerca? **Lucene**.

Ha bisogno di un database XML nativo? **Apache Xindice**.

Ha bisogno di creare un repository RDF interrogabile in SPARQL? **Sesame o ARC**.

Ha bisogno di una piattaforma di computing parallelo e distribuito? **Hadoop**.

Ha bisogno di piattaforme/ambienti di sviluppo software integrati? **Eclipse, Aptana, Seam Framework, ecc.**

Ha bisogno di un advertising server? **OpenX**.

Ha pure bisogno di un PBX? **Asterisk**.

# **Quanto costa avviare un'azienda tecnologica da zero basata sull'opensource e sul cloud computing? - 3**

**I costi riguardanti il potenziale consumo di risorse di calcolo propri dell'utility computing sono destinati a scendere sempre più, sia che si scelga il cloud computing, sia che si scelga il cluster in house.**

Le risorse di calcolo si potranno sempre di più affittare grazie al cloud computing (in alcuni casi sono gratis – Google Apps) a causa dello sviluppo di questa tecnologia e della concorrenza.

I cluster di PC hanno costi sempre più moderati e sono scalabili (basta aggiungere un nodo).

Pro e contro delle due soluzioni:

Affittare - Pro: bassi costi, Contro: scarsa sicurezza dei dati (zero-knowledge)

Cluster in house - Pro: privacy garantita, Contro: costi gestione

# Conclusioni

**Serve “solo” personale specializzato  
e  
idee**

**Non vale la pena provarci ?**

# Cosa centra la Virtualizzazione con il Cloud Computing?

**Il Cloud computing potrebbe anche essere fornito senza l'utilizzo di tecnologie di virtualizzazione.**

- Spesso tuttavia l'utilizzo di tecnologie di virtualizzazione consente di ridurre i costi operativi e in conto capitale.
- Essere in grado di fornire molto rapidamente delle macchine virtuali non è comunque efficiente, se servono diversi mesi per effettuare il provisioning e l'installazione degli host fisici (vedi strumenti di supporto all'installazione come oVirt e Cobbler).
  - Inoltre, *il tempo impiegato per la fornitura dello strato di virtualizzazione è recuperato dai risparmi associati al non dover utilizzare server fisici?*
  - *Importanza di avere tool di installazione (Cobbler), monitoring e accounting il più possibile automatizzati.*

# Virtualizzazione al servizio del Cloud Computing: **Vantaggi**

## **Server consolidation**

- Molte VM su una macchina fisica.
- Riduce i costi di acquisizione dell'hardware e può semplificare operazioni come monitoring e amministrazione.

## **Sandboxing**

- Isolamento delle applicazioni.
- Sviluppo, test e debug di codice.
- Creazione di ambienti per applicazioni legacy.

## **Creazione di VM**

on-demand.

# Virtualizzazione al servizio del Cloud Computing: **Vantaggi**

## **Disaccoppiamento dell'hardware dal software.**

- Suspend/Resume delle VM.
- Migrazione di VM tra server fisici (con vari gradi di sofisticazione).

## **Test di nuove versioni di sistema operativo, applicazioni.**

- O di vecchie versioni: data preservation

## **Emulazione di hardware non presente sulla macchina fisica.**

## **Esecuzione di applicazioni non compatibili con il sistema operativo della macchina fisica.**



# Virtualizzazione al servizio del Cloud Computing:

## **Svantaggi**

### **Sicurezza**

Si mettono sullo stesso hardware diversi sistemi operativi, gestiti da un software - aumento della probabilità di bug e aumento del numero dei cosiddetti attack vectors

- VM-to-VM network attacks.
- VM-to-HV (KVM o XEN). KVM è un modulo del kernel Linux. Xen è direttamente connesso all'hardware. tutto può diventare compromesso
- VM-to-QEMU QEMU è un software grande e complesso. Se viene compromesso, l'attacco può arrivare al sistema operativo.

### **Performance**

- Overhead per la macchina fisica.
- Minori prestazioni per la VM, soprattutto nell'I/O (le cose stanno migliorando)

# Virtualizzazione al servizio del Cloud Computing: **Vantaggi**

- Installazione/reinstallazione di server o di applicazioni su VM di per sé non è Cloud computing
- Verificare con le 5 caratteristiche del Cloud mostrate precedentemente:
  - Self-service, on-demand NO (tipicamente è un dipartimento IT che fornisce le VM)
  - Accesso attraverso la rete NO (deployment limitato a “internal customers”)
  - Pool di risorse Sì
  - Elasticità NO (tipicamente è un dipartimento IT che deve installare sistema operativo + software, e non necessariamente in modo scalabile)
  - Pagamento a consumo NO (spesso il billing non viene fatto a consumo ma in modo tradizionale)

Un esempio di virtualizzazione che non è Cloud ?

(ma che è complementare al Cloud ) **oVirt**

# Virtualizzazione al servizio del Cloud Computing: **oVirt + Openstack**

Nonostante le differenze, ci sono chiaramente punti di sovrapposizione tra framework di virtualizzazione e framework di Cloud computing

Nel mondo open source, un caso di integrazione è dato dalla crescente convergenza tra oVirt e OpenStack