**Comparison**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Properties** | **Amazon EC2** | **Google AppEngine** | **Microsoft Azure** | **Manjrasoft Aneka** |
| Service Type | IaaS | IaaS – PaaS | IaaS – PaaS | PaaS |
| Support for (value offer) | Compute/storage | Compute (web applications) | Compute | Compute |
| Value added service provider | Yes | Yes | Yes | Yes |
| User access interface | Web APIs and  Command Line Tools | Web APIs and  Command Line Tools | Azure Web Portal | Web APIs, Custom GUI |
| Virtualization | OS on Xen Hypervisor | Application Container | Service Container | Service Container |
| Platform (OS & runtime) | Linux, Windows | Linux | .NET on Windows | .NET/Mono on  Windows, Linux, MacOS X |
| Deployment model | Customizable VM | Web apps (Python, Java, JRuby) | Azure Services | Applications (C#, C++, VB, ….) |
| If PaaS, ability to  deploy on 3rd party IaaS | N.A. | No | No | Yes |

**General information**

| **Software** | **Written in** | **As a service** | **Local installations** |
| --- | --- | --- | --- |
| [**Cloud.com**](http://en.wikipedia.org/wiki/Cloud.com)**/ CloudSim** | Java, C | Yes | ? |
| [**Eucalyptus (computing)**](http://en.wikipedia.org/wiki/Eucalyptus_%28computing%29) | Java, C | Yes | ? |

**Supported Hosts**

(what the cloud software runs on)

| **Software** | **Linux** | **Windows** |
| --- | --- | --- |
| [**Cloud.com**](http://en.wikipedia.org/wiki/Cloud.com) **/ CloudSim** | Yes | ? |
| [**Eucalyptus (computing)**](http://en.wikipedia.org/wiki/Eucalyptus_%28computing%29) | ? | ? |

**Supported Clients**

(what the cloud software will run as a virtual instance)

| **Software** | **Linux** | **Windows** | **VMware** | **Xen** | **KVM** | **VirtualBox** |
| --- | --- | --- | --- | --- | --- | --- |
| [**Cloud.com**](http://en.wikipedia.org/wiki/Cloud.com) **/ CloudSim** | Yes | ? | Yes | Yes | Yes | Yes |
| [**Eucalyptus (computing)**](http://en.wikipedia.org/wiki/Eucalyptus_%28computing%29) | ? | ? | Yes | Yes | Yes | ? |

**Providers**

(prominent hosting providers who offer as SaaS)

| **Software** | **Amazon EC2** | **Rackspace** | **GeoGrid** | **Other** |
| --- | --- | --- | --- | --- |
| [**Cloud.com**](http://en.wikipedia.org/wiki/Cloud.com) | ? | ? | ? | itself |
| [**Eucalyptus (computing)**](http://en.wikipedia.org/wiki/Eucalyptus_%28computing%29) | ? | ? | ? | itself |

**Features**

* Failover - supports automatic handling of hardware failures ([Failover](http://en.wikipedia.org/wiki/Failover)) (partial indicates failovers of controller nodes are not supported)
* OCCI - supports [Open Cloud Computing Interface](http://en.wikipedia.org/wiki/Open_Cloud_Computing_Interface)
* vCloud - supports [vCloud](http://en.wikipedia.org/wiki/VCloud) migration
* S3 - supports [Amazon S3](http://en.wikipedia.org/wiki/Amazon_S3) volumes

| **Software** | **Failover** | **OCCI** | **vCloud** | **S3** |
| --- | --- | --- | --- | --- |
| [**Cloud.com**](http://en.wikipedia.org/wiki/Cloud.com) **/ CloudStack** | Partial | ? | ? | ? |
| [**Eucalyptus (computing)**](http://en.wikipedia.org/wiki/Eucalyptus_%28computing%29) | No | ? | ? | Yes |

<http://setandbma.wordpress.com/2010/01/21/cloud-economics-a-platform-comparison/>

<http://kalirajanl.wordpress.com/2011/05/16/comparison-of-open-source-cloud-platforms/>

<http://searchcloudcomputing.techtarget.com/feature/Comparing-open-source-cloud-platforms-OpenStack-versus-Eucalyptus>

Any new technology adoption happens because of one of the three reasons:

1. **Capability:** It allows us to do something which was not feasible earlier
2. **Convenience:** It simplifies
3. **Cost:** It significantly reduces cost of doing something

Our expectation from cloud computing is **cost saving** … (1) through elastic capacity and (2) through economy of scale. So, for any CIO interested in moving to cloud, it is very important to understand what the cost elements are for different cloud solutions.

# Amazon:

* **Overview**: You can create one or more instances of a virtual machine for processing and for storage
  + You pay based on time the instances are running and not on how much they are used – **if an instance is idle, you still pay for it**
  + There are three physically **different locations** where the facility is available (called availability zones) – **US**(N. Virginia, N. California) and **EU**(Ireland)
  + When you either shutdown the machine instance or it crashes for whatever reason **you lose all your data** – to preserve your data you need to opt for one of the 4 possible data persistence mechanisms
  + It is possible to have a **reserve instance** (for 1 year or 3 years) for an initial payment and discounted rate of usage – however, I do not think it provides any guarantee against data loss because of machine crash
  + Data storage can be **both relational and non-relational**
* **Machine Instance:** Virtual machine can be of different capacity – **Standard**(Small, Large, Extra Large), **High-Memory**(Double Extra Large, Quadruple Extra Large), **High-CPU**(Medium, Extra Large)
  + [***Charge for Machine Usage***](http://aws.amazon.com/ec2/)**:** You are charged for the time you keep the instance of the machine running – the time is **calculated in hours**, any fraction of hour is taken as full hour
    - **Hourly charge** vary from **$0.085** (Small – Linux – N. Virginia) to **$3.16** (Quadruple Extra Large – Windows – N. California)
    - Both **Linux** and **Windows** machine instances are supported – Windows machines are about **40% more expensive** – other software charges are extra
  + There are separate **charges** for mapping **IP addresses**, for **monitoring & auto scaling** ($0.015 per instance per hour) and **load balancing**
  + A message queue is available (Simple Queue Service – **SQS**) but again it has a separate charge – **$0.1 to $0.17 per GB** depending on the total monthly volume
* **Data Persistence:** To persistent data storage you can one of the 4 alternatives – Elastic Block Store (**EBS**) **Simple DB**, Simple Storage Service (**S3**) or Relational Database Service (**RDS**)
  + **EBS** behaves like a mountable hard drive and is automatically replicated. You can use it in any way you want.
  + **Simple DB and S3** storage mechanism is **not RDBMS** – that is you do not have tables therefore you cannot retrieve records through using JOIN
  + **RDS** is an instance of MySQL – so you can use it like a **normal RDBMS**
  + [***Charges for EBS***](http://aws.amazon.com/ebs/): you pay for allocated storage and I/O
    - **Storage** charge are **$0.10** per allocated GB per month
    - **I/O requests** are charged separately at **$0.10** per million I/O
  + [***Charges for Simple DB***](http://aws.amazon.com/simpledb/): you pay separately for CPU, disk space and data transfer – though up to a limit they are **free** (25 CPU hours, 1GB data transfer, 1GB of storage)
    - **CPU** usage calculation is normalized to 1.7 GHz Xeon (2007) processor and works out to **$0.14 to $0.154 per hour** depending on location
    - **Data transfer** In is free till June 2010 and charge for transfer Out is between **$0.1 to $0.17**  
      **per GB** depending on the total monthly volume
    - Actual **storage** is charger at **$0.25 to $0.275 per GB per month** – it includes 45 bytes of overhead for each item uploaded
  + [***Charges for S3***](http://aws.amazon.com/s3/)**:** You are charged for **disk space, data transfer and number of request made** instead of CPU usage – data transfer charges are the same
    - **Storage** charge varies from **$.055 to $0.165** per GB per month making it slightly cheaper than Simple DB but at a higher level of usage (more than 1000 TB)
    - **I/O requests** are charged separately – you pay between **$0.01 to $0.011 per 1,000 write** requests and **$0.01 to $0.011 per 10,000 read** requests – deletes are free
  + [***Charge for RDS***](http://aws.amazon.com/rds/): You pay **for storage, I/O request, data transfer and machine instance** (Small, Large, Extra Large, Double Extra Large, Quadruple Extra Large) based on usage
    - You pay for **RDS instance** – charges vary from **$0.11 to $3.10 per hour** depending on the instance size
    - The **storage** charge is not pay as you use – you have to decide in advance (5 GB to 1 TB) and the charges are **$0.10 per GB per month**
    - The is no charge for **backup up** to the amount of storage you have chosen but you have to pay **$0.15 per GB per month** for extra backup
    - You pay separately for **I/O** at **$0.10 per 1 million I/O requests**

# Google:

* **Overview**: Application written in **Python** or **Java** can directly be deployed – the implementation is a subset
  + **No** need to instantiate any **virtual machine**
  + You are **charged** on the actual normalized **CPU cycles used**
  + Storage is **only non-relational**
  + **Charge** is calculated on these parameters – **bandwidth, CPU, storage, emails send**
  + You have [***free quota***](http://code.google.com/appengine/docs/quotas.html) for each of these parameters – it is enough for development, testing and small deployment
  + There are **limits** imposed for **peak** usage on many different parameters – with daily limits & limits on usage in a burst
  + You will need to rewrite your application to work on Google App Engine – [***see this***](http://setandbma.wordpress.com/2009/08/24/does-cloud-require-change-in-programming-mindset/)
  + [***Charge for CPU usage***](http://code.google.com/appengine/docs/billing.html): It is calculated in CPU seconds equivalent to 1.2 GHz Intel x86 processor
    - You pay **$0.10 per hour** of **CPU** usage for processing requests
    - **6.5 hours** of CPU time is **free**
    - You **do not pay** for CPU **idle** time
  + [***Charge for storage***](http://code.google.com/appengine/docs/billing.html): Only non-relational storage is available
    - You pay **$0.15 per GB per month** – the size includes overhead, metadata and storage required for indexes
    - It **includes** data stored in the **datastore, memcache, blobstore**
    - You pay for **CPU** usages for data I/O at **$0.10 per hour**
    - **60 hours** of CPU time for data I/O is **free**
    - Up to **1 GB** of storage is **free** – [FAQ page](http://code.google.com/appengine/kb/billing.html) says that it is 500 MB
    - You are charged every day at **$0.005 GB per day** after subtracting your free quota
  + [***Charge for bandwidth usage***](http://code.google.com/appengine/docs/billing.html): Inward and outward bandwidth usage is charged at different rate
    - You pay **$0.10 per GB** for **incoming** traffic
    - You pay **$0.12 per GB** for **outgoing** traffic
    - **1 GB** of incoming traffic and **1 GB** of outgoing traffic is **free**

# Microsoft:

* **Overview**: Offering has 3 main parts **– Windows Azure, SQL Azure and App Fabric**
  + Details available on the Microsoft site is **more** about the **vision** of the product **than** about what is implemented **here and now**.
  + However this document “[***Introducing Windows Azure***](http://download.microsoft.com/download/0/8/7/087A3AE1-2880-4452-88DD-09398D0A522A/Introducing_Windows_Azure.doc)” is good
  + It uses [***Hyper-V***](http://en.wikipedia.org/wiki/Hyper-V) for virtualization – it works more like Amazon than like Google
  + There is an introductory offer where the service can be avail for free
  + The **development** environment is **Visual Studio** through an SDK
  + The **emphasis** of creating applications which partly runs in **premise**  
    **and** partly on **cloud**
  + Microsoft wants to keep the programming model as much unaltered as possible – [***see this***](http://setandbma.wordpress.com/2009/10/01/cloud-does-not-require-change-in-programming-model-microsoft/)
  + [***Charge for CPU usage***](http://www.microsoft.com/windowsazure/pricing/): It is calculated in CPU seconds equivalent to 1.2 GHz Intel x86 processor
    - You pay **$0.12 per hour** of **CPU** usage for processing requests
  + [***Charge for storage***](http://www.microsoft.com/windowsazure/pricing/): Only relational storage is available in form of SQL Azure
    - You pay **$0.15 per GB per month**
    - **Storage transactions** are charged separately at **$0.01 per 10,000 transactions**
  + [***Charge for bandwidth usage***](http://www.microsoft.com/windowsazure/pricing/): Inward and outward bandwidth usage is charged at different rate
    - You pay **$0.10 per GB** for **incoming** traffic – rates for **Asia** are different **$0.30 per GB**
    - You pay **$0.15 per GB** for **outgoing** traffic – rates for **Asia** are different **$0.45 per GB**