

Cloud Platforms

Various types and their properties

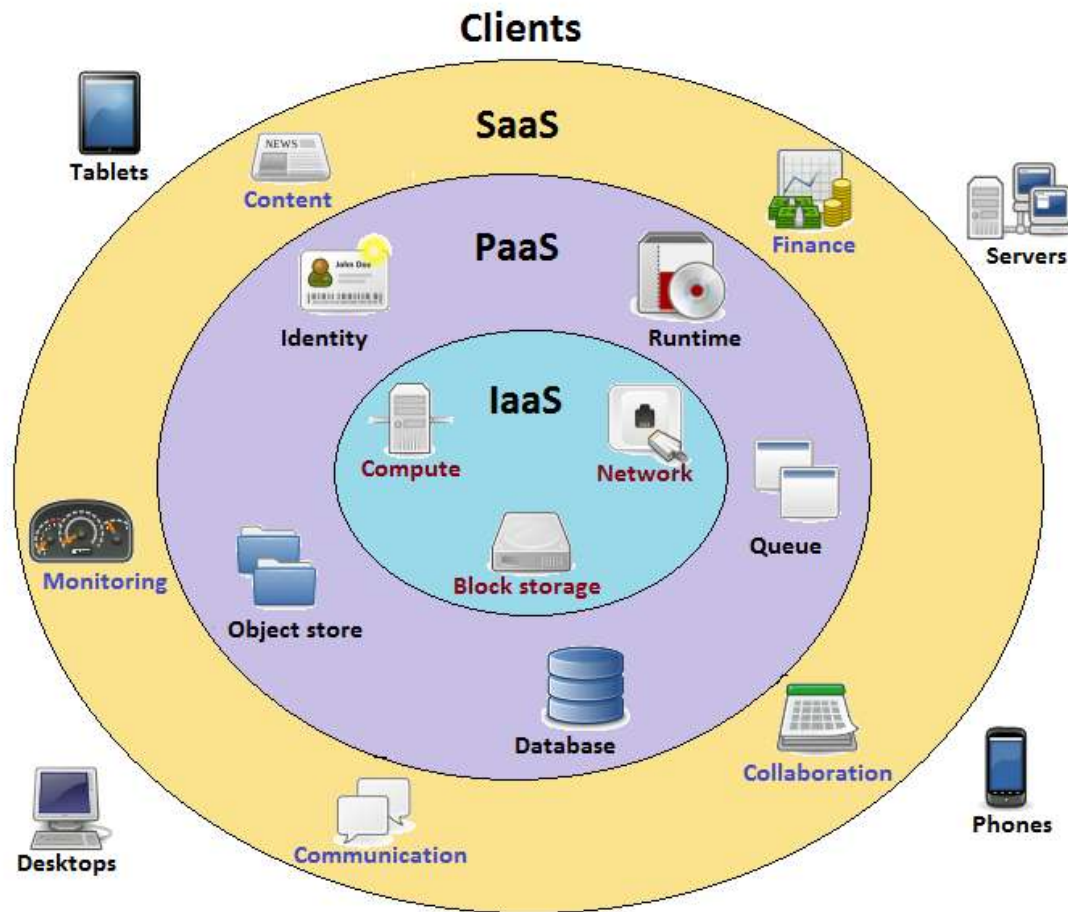
Prof. Balwinder Sodhi

Computer Science and Engineering, IIT Ropar

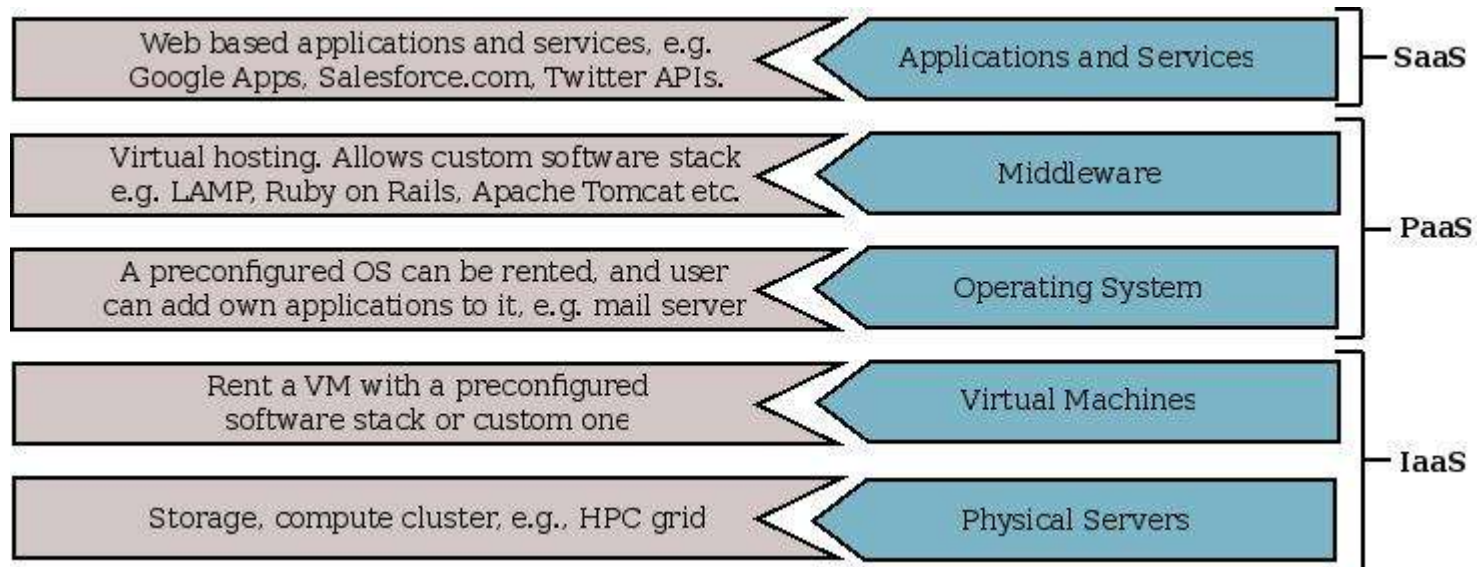
Cloud Classification

- Service model based
 - Depends on the cloud services being offered
 - Infrastructure as a Service (IaaS), e.g., AWS EC2
 - Platform as a Service (PaaS), e.g., Google App Engine
 - Software as a Service (SaaS), e.g., Salesforce.com
- Deployment model based
 - Depends on how a cloud is setup/deployed
 - Private → Operated by and for an individual entity
 - Public → Available to general public like a utility
 - Hybrid → Private and public connected together
 - Community → Setup by and for a group having shared goals

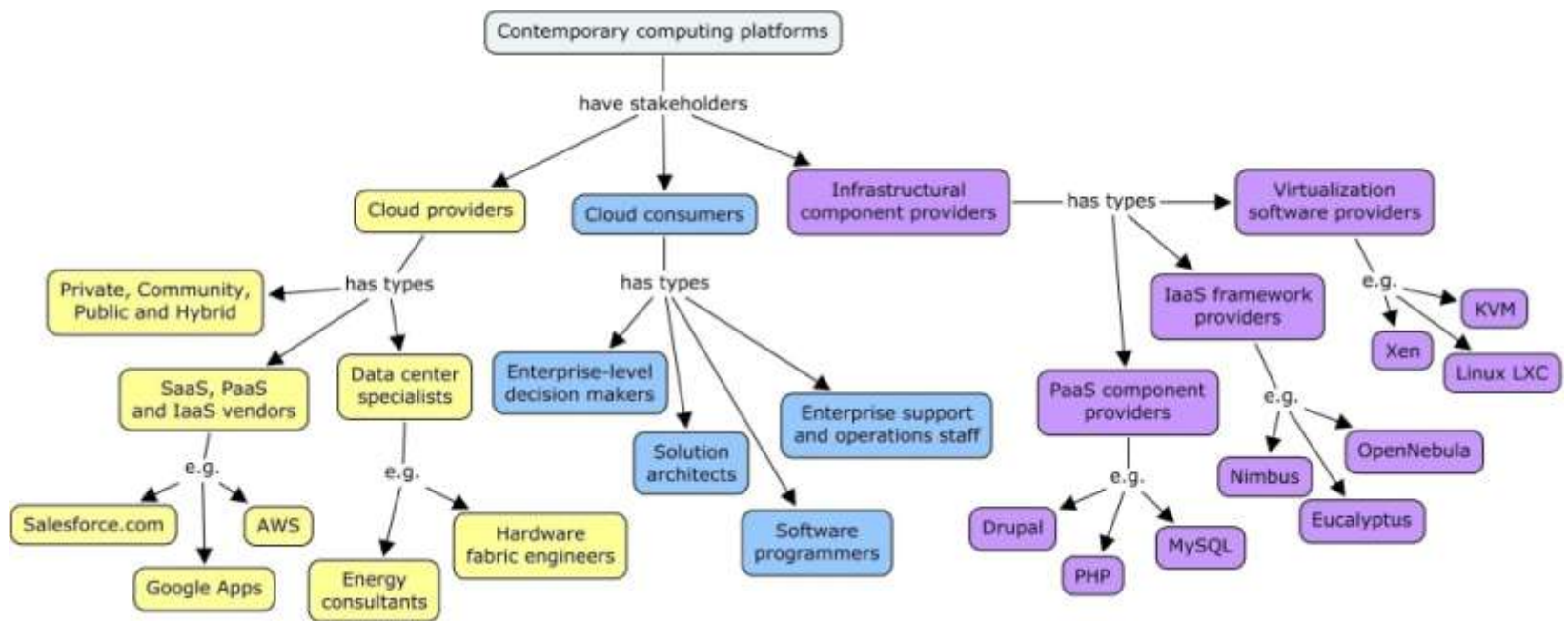
Logical View of Cloud Computing



Logical View of Cloud Computing



Various Stakeholders In Cloud



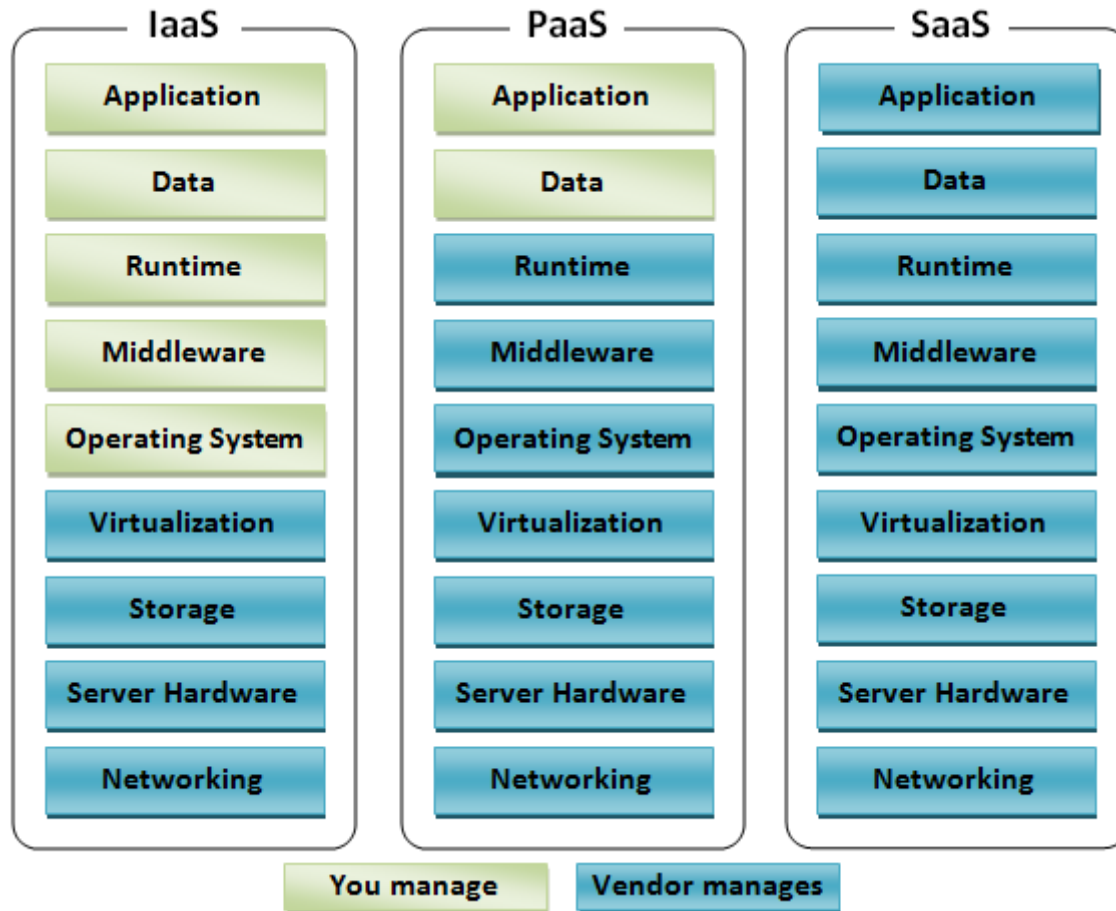
Variants of Cloud Platforms

Characteristics Common To All Variants

- **Programmatic and Self-service provisioning of resources**
- Multi-tenancy → Shared underlying computing infrastructure
- **Lack of absolute control/custody of data and computing assets**
- Computing as a utility accessible over the network
- **Measured service**
- Political/legal/geographic location can be transparent to clients
- **Different structure for software licensing**
- Potential to abuse the relative anonymity behind registration and usage models

Service model based

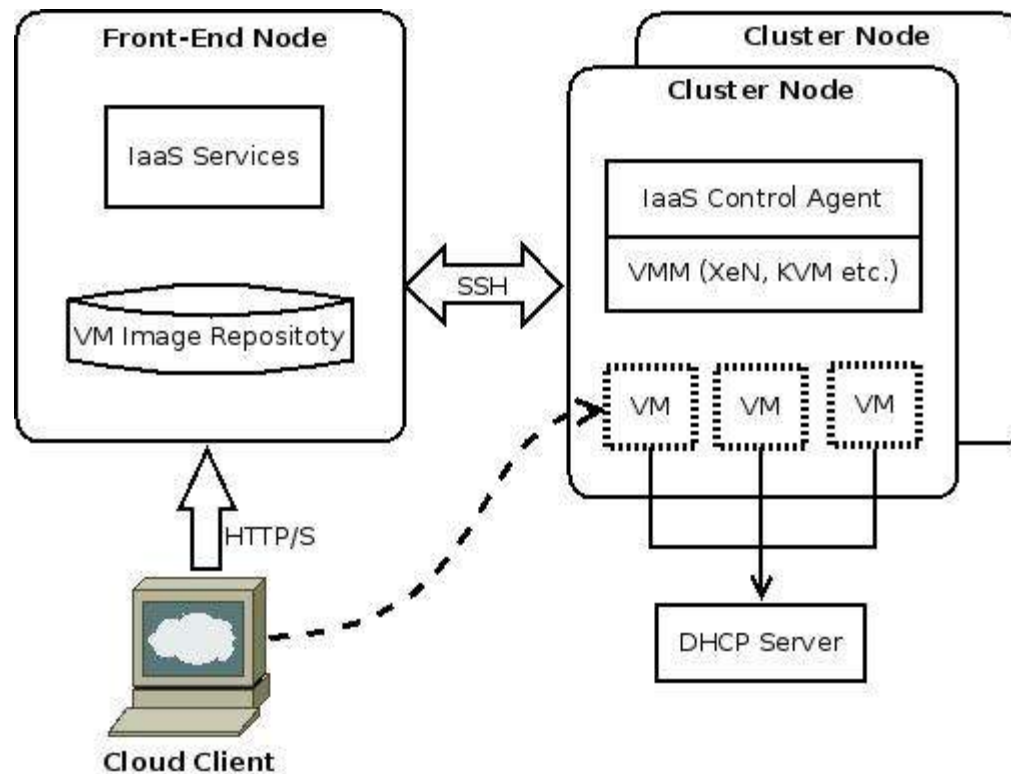
IaaS/PaaS/SaaS Side-by-Side



Infrastructure as a Service (IaaS)

- Provides fundamental computing resources
 - Processing
 - Storage
 - Networks
 - Etc.
- User can deploy and run arbitrary software
 - E.g. operating systems and applications
- No control of underlying hardware
 - Can allow limited control of networking components
 - Full control of OS
- Typically, enabled via virtualization technologies
 - VM is a common deployment unit
 - E.g. AWS EC2

IaaS Architecture



IaaS Cloud Characteristics

- Provides bare-bones computing infrastructure
 - Storage, compute networking etc., often via a VM
- Cloud user responsible for installing/managing all software on VM
- Allows resource utilization monitoring and reacting to events
 - Responsibility again lies with the user application
- Limited control on networking components, e.g. host firewalls
- By far the most flexible cloud variant
 - User can configure/control the VM and software stack
 - This also means more effort from the cloud user

Vendor Example | Amazon Web Services

- A leading public IaaS cloud provider
- Offers wide variety of services
 - Compute (EC2)
 - Storage (S3)
 - Databases (RDS, SimpleDB etc.)
 - DNS system (Route 53)
- Easy to sign-up for an account
 - Requires account verification (usually via phone)

Amazon Web Services (AWS)

The screenshot shows the Amazon EC2 Console Dashboard for the US West (Oregon) region. The URL in the browser is <https://console.aws.amazon.com/ec2/home?region=us-west-2#>. The page features a navigation sidebar on the left, a main content area with a 'Getting Started' section, and a 'My Resources' summary on the right.

Annotations:

- A box labeled "Select hosting region" points to the "Region:" dropdown menu in the navigation sidebar, which is currently set to "US West (Oregon)".
- A box labeled "Click here to start VM creation wizard" points to the "Launch Instance" button in the "Getting Started" section.

Getting Started Section:

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

[Launch Instance](#)

Note: Your instances will launch in the US West (Oregon) region.

Service Health:

Current Status	Details
✔ Amazon EC2 (US West - Oregon)	Service is operating normally

[View complete service health details](#)

My Resources Summary:

- 1 Running Instance
- 0 Elastic IPs
- 1 EBS Volume
- 0 EBS Snapshots
- 1 Key Pair
- 0 Load Balancers
- 0 Placement Groups
- 2 Security Groups

Events:

US West (Oregon): No events

Related Links:

© 2008 - 2012, Amazon Web Services LLC or its affiliates. All rights reserved. | [Feedback](#) | [Support](#) | [Privacy Policy](#) | [Terms of Use](#) | An [amazon.com](#) company

Select Machine Image

Request Instances Wizard Cancel X

CHOOSE AN AMI INSTANCE DETAILS CREATE KEY PAIR CONFIGURE FIREWALL REVIEW

Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its **Select** button.

Quick Start My AMIs Community AMIs

	LAMP Web Starter (AMI Id: ami-2cb05345) Fedora Core 8, 32-bit architecture, PHP 5.2, Apache 2.2, and MySQL 5	Select
	Basic Fedora Core 8 (AMI Id: ami-84db39ed) Minimal Fedora Core 8, 32-bit architecture, and Amazon EC2 AMI Tools.	Select
	Basic 64-bit Fedora Core 8 (AMI Id: ami-86db39ef) Fedora Core 8, 64-bit architecture, and Amazon EC2 AMI tools.	Select
	Getting Started on Microsoft Windows Server 2008 (AMI Id: ami-c5e40dac) Microsoft Windows Server 2008 R1 SP2 Datacenter edition, 32-bit architecture, Microsoft SQLServer 2008 Express, Internet Information Services 7, ASP.NET 3.5.	Select
	Basic Microsoft Windows Server 2008 (AMI Id: ami-c3e40daa) Microsoft Windows 2008 R1 SP2 Datacenter edition and 32-bit architecture.	Select
	Basic 64-bit Microsoft Windows Server 2008 (AMI Id: ami-d9e40db0) Microsoft Windows 2008 R1 SP2 Datacenter edition and 64-bit architecture.	Select

Select Machine Type

Request Instances Wizard

CHOOSE AN AMI | **INSTANCE DETAILS** | CREATE KEY PAIR | CONFIGURE FIREWALL | REVIEW

Provide the details for your instance(s). You may also decide whether you want to launch your instances as "on-demand" or "spot" instances.

Number of Instances: Availability Zone:

Instance Type:

Type	CPU Units	CPU Cores	Memory
Micro (t1.micro)	Up to 2 ECUs	1 Core	613 MB
Large (m1.large)	4 ECUs	2 Cores	7.5 GB
Extra Large (m1.xlarge)	8 ECUs	4 Cores	15 GB
High-Memory Extra Large (m2.xlarge)	6.5 ECUs	2 Cores	17.1 GB
High-Memory Double Extra Large (m2.2xlarge)	13 ECUs	4 Cores	34.2 GB
High-Memory Quadruple Extra Large (m2.4xlarge)	26 ECUs	8 Cores	68.4 GB
High-CPU Extra Large (c1.xlarge)	20 ECUs	4 Cores	15 GB

Back

© 2008 - 2009, Amazon Web Services LLC or its affiliates. All right reserved. Feedback Support Privacy Policy Terms of Use An amazon.com company

Pick the resources you need for VM.
Cost varies with instance type *:
Small (Default): \$0.080/Hour
Medium: \$0.160/Hour
Large: \$0.320/Hour
Extra Large: \$0.640/Hour

Configure Network Access

Request Instances Wizard

CHOOSE AN AMI | INSTANCE DETAILS | CREATE KEY PAIR | **CONFIGURE FIREWALL** | REVIEW

Security groups determine whether a network port is open or blocked on your instances. You may use an existing security group, or we can help you create a new security group to allow access to your instances using the suggested ports below. Add additional ports now or update your security group anytime using the Security Groups page. All changes take effect immediately.

Choose one or more of your existing Security Groups

Create a new Security Group

1. Name your Security Group

2. Describe your Security Group

3. Define allowed Connections

Application	Transport	Port	Source Network (IPv4 CIDR)	Actions
SSH	tcp	22	All Internet	<input type="button" value="Remove"/>
Select...	-	-	All Internet Change	<input type="button" value="Add Rule"/>

To allow us to SSH into this VM

We Have Our Machine Running!

The screenshot displays the AWS Management Console interface for the 'My Instances' page. The browser address bar shows the URL: <https://console.aws.amazon.com/ec2/home?region=us-west-2#s=Instances>. The user is identified as 'Balwinder Sodhi'.

Navigation Panel: Region: US West (Oregon). Menu items include EC2 Dashboard, Events, INSTANCES (Instances, Spot Requests, Reserved Instances), IMAGES (AMIs, Bundle Tasks), ELASTIC BLOCK STORE (Volumes, Snapshots), and NETWORK & SECURITY (Security Groups, Elastic IPs, Placement Groups, Load Balancers, Key Pairs, Network Interfaces).

My Instances Table:

Name	Instance	AMI ID	Root Device	Type	State	Status Checks	Alarm Status	Mo
<input checked="" type="checkbox"/>	LMS Server	i-914cd9a2	ami-4438b474	ebs	m1.medium	running	2/2 checks p	none

1 EC2 Instance selected.

EC2 Instance: LMS Server (i-914cd9a2)
ec2-54-245-0-9.us-west-2.compute.amazonaws.com

Description | Status Checks | Monitoring | Tags

AMI: ubuntu/images/ebs/ubuntu-precise-12.04-amd64-server-20120616 (ami-4438b474) *none*

Zone: us-west-2a

Type: m1.medium

Scheduled Events: No scheduled events

VPC ID: -

Security Groups: quicklaunch-1. [view rules](#)

State: running

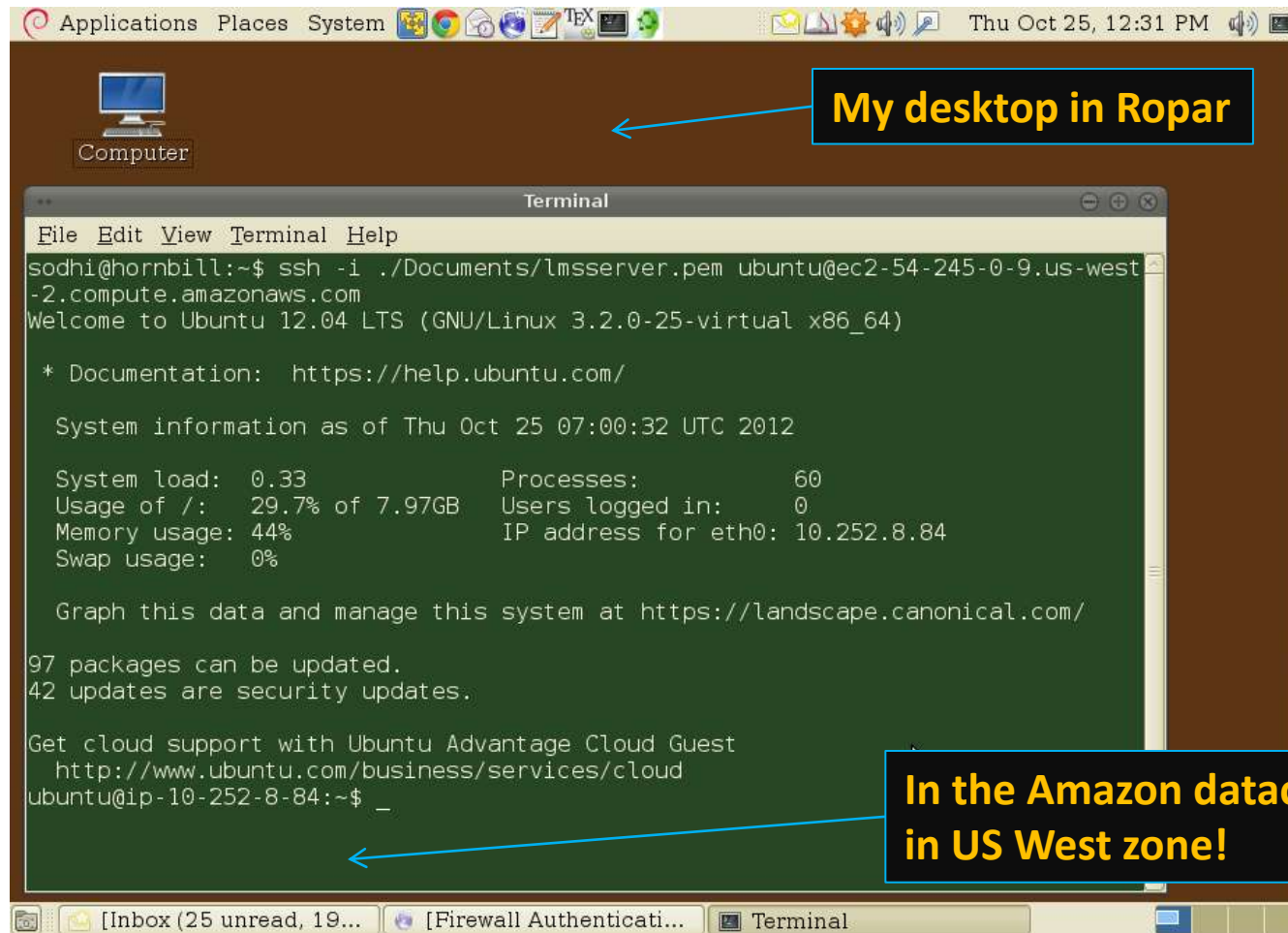
Owner: 713326102970

Subnet ID: -

Annotations: 'Our instance' points to the instance name 'LMS Server' in the table. 'DNS name' points to the instance's public IP address 'ec2-54-245-0-9.us-west-2.compute.amazonaws.com'.

© 2008 - 2012, Amazon Web Services LLC or its affiliates. All rights reserved. | [Feedback](#) | [Support](#) | [Privacy Policy](#) | [Terms of Use](#) | An [amazon.com](#) company

SSH Into Our VM



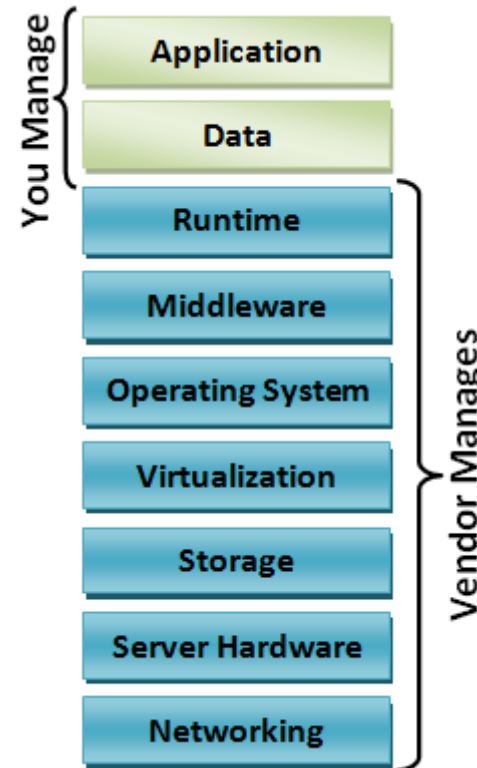
Real-time Billing Details

Details			
Expand All Services Collapse All Services			Printer Friendly Version
AWS Service Charges			\$92.72
<input type="checkbox"/> Amazon Elastic Compute Cloud Download Usage Report >>			\$92.72
US West (Oregon) Region			
Amazon EC2 running Linux/UNIX			
\$0.160 per Medium Instance (m1.medium) instance-hour (or partial hour)	575 Hrs		92.00
Amazon EC2 EBS			
\$0.10 per GB-month of provisioned storage	6,161 GB-Mo		0.62
\$0.10 per 1 million I/O requests	1,006,423 IOs		0.10
<input type="checkbox"/> AWS Data Transfer (excluding Amazon CloudFront)			\$0.00
\$0.000 per GB - data transfer in per month	0.023 GB		0.00
\$0.000 per GB - first 1 GB of data transferred out per month	0.032 GB		0.00
<input type="checkbox"/> VAT to be collected			\$0.00
<small>† Usage and recurring charges for this statement period will be charged on your next billing date, November 1, 2012. Estimated charges shown</small>			

Platform as a service

Platform as a Service (PaaS)

- NIST* definition: “... deploy onto the cloud infrastructure consumer-created or acquired applications created **using programming languages, libraries, services, and tools supported by the provider**”
- Consumer responsible only for writing application code
- Vendor gives sandboxed environment to develop/deploy applications
- Multiple consumers share the platform

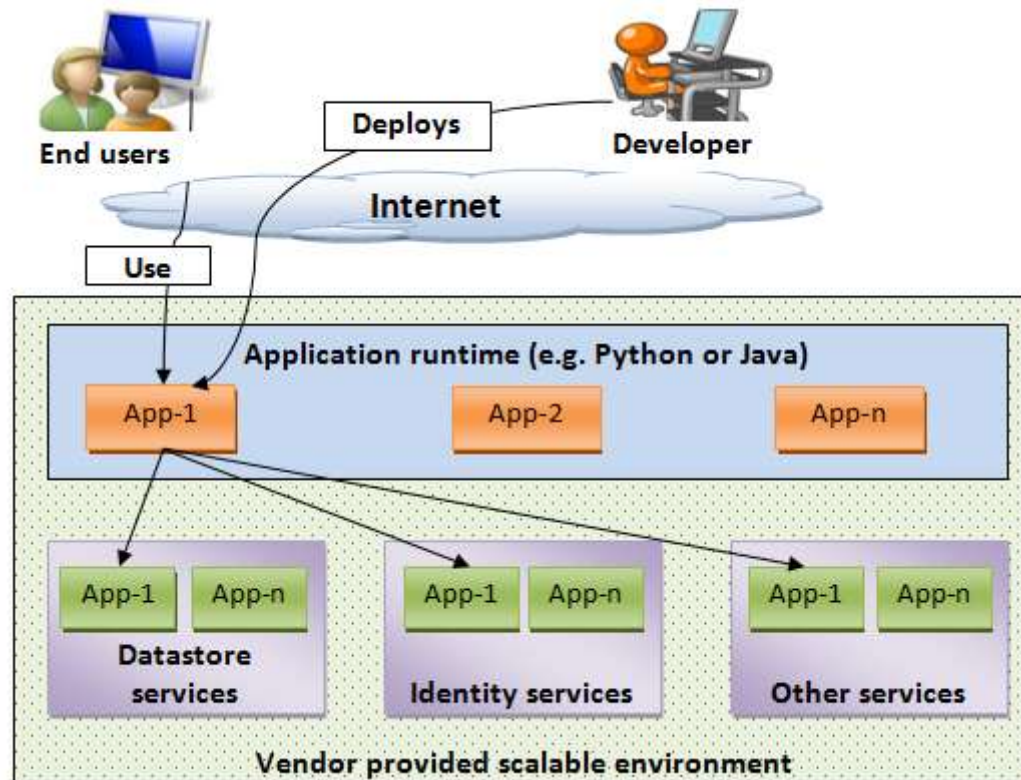


*NIST = National Institute of Standards and Technology
Computer Science and Engineering, IIT Ropar

PaaS Cloud Characteristics

- Allows only provider supported programming languages, tools, APIs and components for building applications
- No control of underlying infrastructure
 - Network, servers, operating systems, or storage
- Can only control deployed application and possibly its hosting environment configurations
- Effort needed to setup/management is lower than IaaS
 - But at the cost of flexibility

PaaS Architecture



Google App Engine (GAE)

- A leading PaaS cloud available to public
- Offers several services to developers
- Has faster ramp-up time to build applications



What Does It Offer

- Lets you run web applications on Google's infrastructure
 - No servers to maintain for you
 - You can focus on your application
- Supports writing apps in several programming languages
 - Java, Python, Go
- You only pay for what you use
 - No set-up costs and no recurring fees
 - Large free quotas for apps

Some GAE Features

- Serve data driven dynamic web apps
- Variety of data storage options
 - Allow queries and transactions
- Automatic scaling and load balancing
- Google Accounts APIs for authentication
- Local development environment
 - Simulates GAE locally on your machine
- Task queues and scheduled tasks
 - Perform work outside the scope of a web request

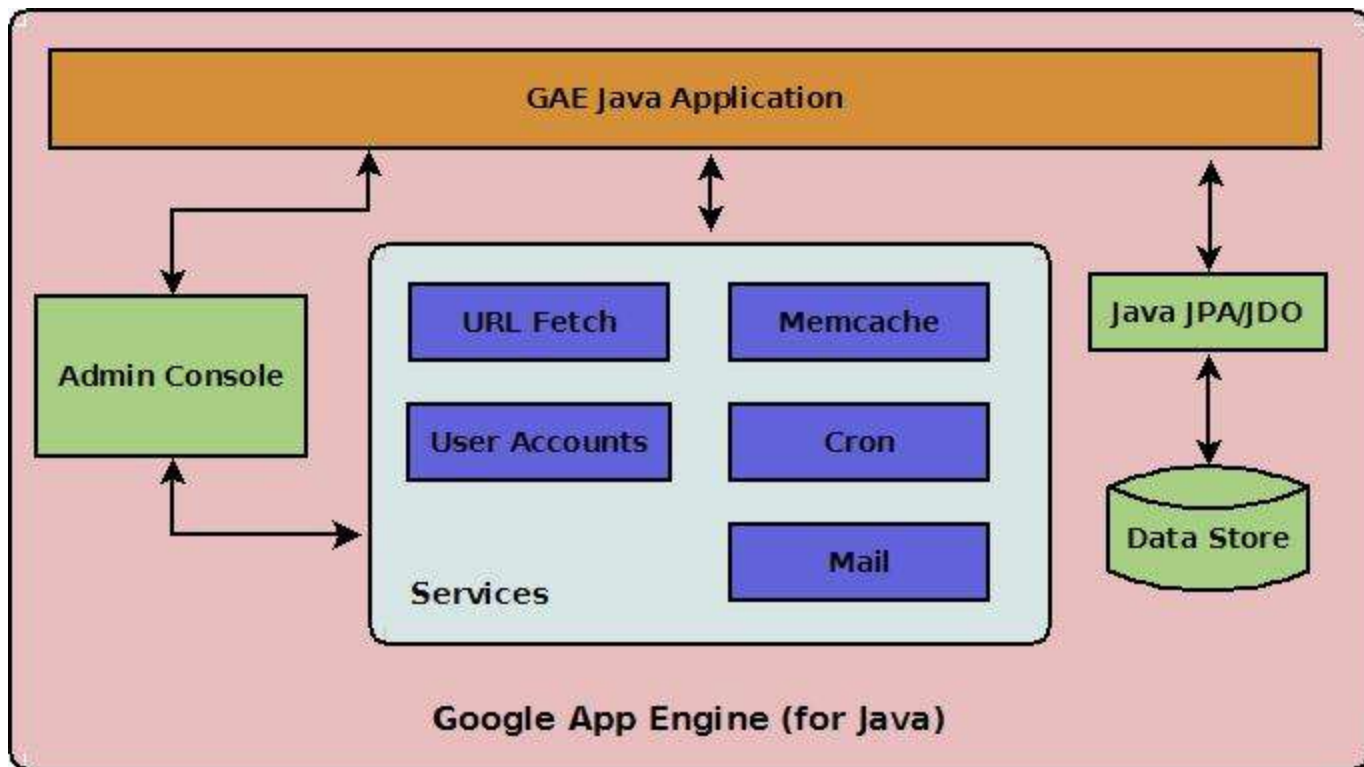
Application Hosting Environment

- Applications run in a secure sandbox environment
 - Limited access to the underlying operating system
 - Allows GAE to load balance requests for application across multiple servers, and automatically scale the servers
 - But some restrictions apply

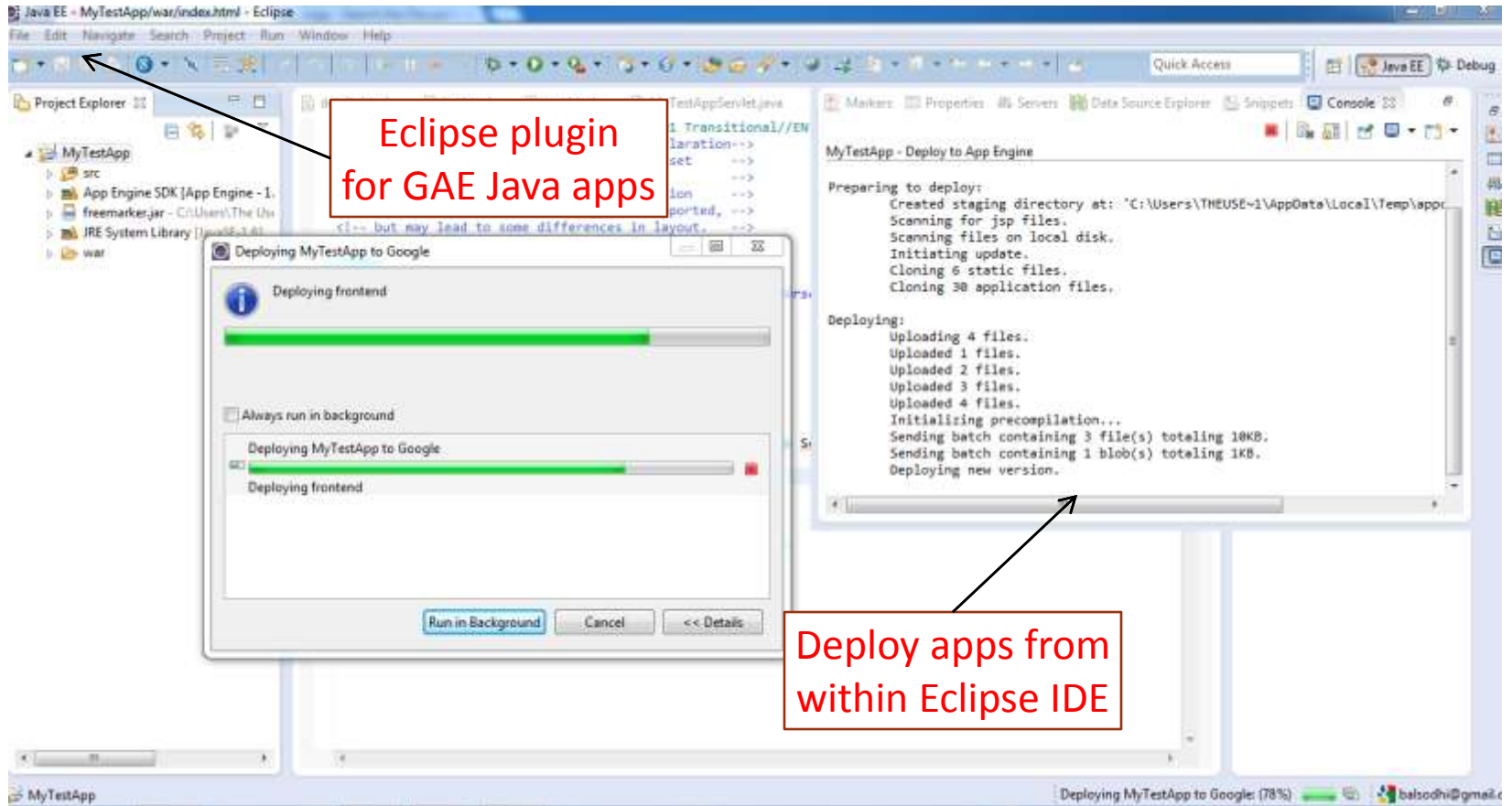
Sandbox Restrictions

- An app can only access other computers on the Internet through the provided URL fetch and email services
- Other computers can only connect to the application via HTTP (S) requests on standard ports
- Cannot write to the file system
- Can read only files bundled with application code
- Application code only runs in response to a web request, a queued task, or a scheduled task
 - Must return response data within 60 seconds in any case
- Cannot spawn a sub-process or execute code after the response has been sent

GAE Architecture



Creating & Deploying GAE Apps



Manage Apps Via GAE Dashboard

The screenshot shows the Google App Engine dashboard for an application named 'searchtheboard'. The browser address bar shows the URL <https://appengine.google.com/dashboard?&...>. The page title is 'Google app engine'. The application name is 'searchtheboard [High Replication]'. The version is '1'. The dashboard includes a navigation menu on the left with sections 'Main' and 'Data'. The 'Main' section includes links for Dashboard, Instances, Logs, Versions, Backends, Cron Jobs, Task Queues, and Quota Details. The 'Data' section includes links for Datastore Indexes, Datastore Viewer, Datastore Statistics, Blob Viewer, Prospective Search, Text Search, Datastore Admin, and Memcache Viewer. The main content area features a 'Charts' section with a line graph showing 'Requests/Second' over time. The graph has a dropdown menu for the chart type and buttons for different time ranges: 6 hrs, 12 hrs, 24 hrs, 2 days, 4 days, 7 days, 14 days, and 30 days. The x-axis shows time points: -1d, -18hr, -12hr, and -6hr. The y-axis shows values from 0.000 to 0.000. Below the chart is an 'Instances' table with columns for 'Number of Instances - Details', 'Average QPS', 'Average Latency', and 'Average Memory'. The table shows 1 total instance with an average QPS of 0.083, average latency of 89.0 ms, and average memory of 54.4 MBytes. Below the table is a 'Billing Status' section showing 'Free - Settings' and 'Quotas reset every 24 hours. Next reset: 17 hrs'. At the bottom is a 'Resource Usage' table with columns for 'Resource' and 'Usage'. The table shows 'Frontend Instance Hours' with a usage of 0% (0.04 of 28.00 Instance Hours).

Lists GAE apps of the user

Options for various management tasks

Detailed app statistics

Number of Instances - Details	Average QPS	Average Latency	Average Memory
1 total	0.083	89.0 ms	54.4 MBytes

Resource	Usage
Frontend Instance Hours	0% 0.04 of 28.00 Instance Hours

Fine-grained Data Store Stats

Display statistics for:

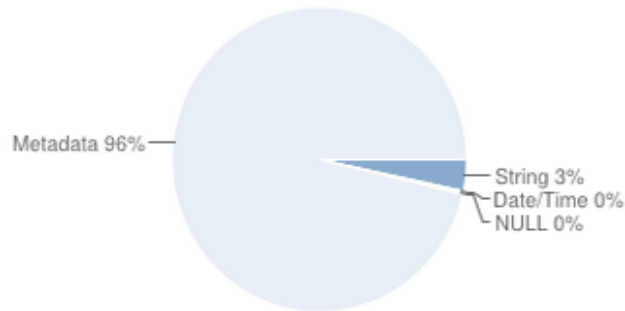
Kind: All Entities ▾

Statistics are updated at least once per day. [Learn more](#)

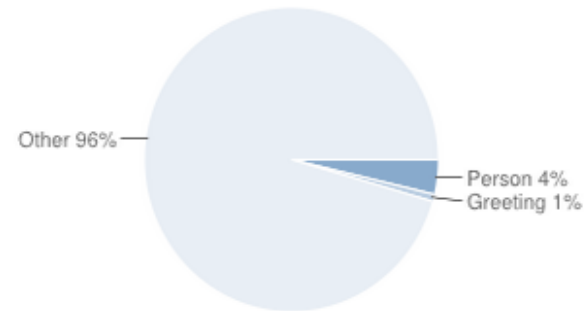
Last updated: 22:47:19 ago

Total Size:	Entities 9 KBytes	Built-in Indexes 81 KBytes	Composite Indexes 0 Bytes	Total 91 KBytes
Entry Count:	30	493	0	

Storage Space by Property Type



Storage Space by Entity Kind



Breakdown by Property Type

Property Type	Size	Index Size
String	269 Bytes	3 KBytes
Date/Time	21 Bytes	176 Bytes
NULL	12 Bytes	162 Bytes
Metadata	87 KBytes	

Billing Information

The screenshot displays the Google App Engine Billing Information page for the application 'searchtheboard [High Replication]'. A red box highlights the 'Details of Resources used by app' section, which is a table showing resource usage for the event 'Usage Report for 2012-10-25' on 2012-10-26. The table includes columns for Resource, Used, Free, Billable, and Charge. A red arrow points from the text box to the 'Resource' column header.

Application: searchtheboard [High Replication]

Below is an event log of all billing-related events for this application. [Download Usage Reports \(CSV\)](#) [Print](#) [Expand All](#)

Events: All Show: 20 Display [Prev 20](#) **1-20** [Next 20](#)

Date	Event	Amount	Balance		
2012-10-26 14:53:17	Usage Report for 2012-10-25		\$0.00		
	Resource	Used	Free	Billable	Charge
	Frontend Instance Hours \$0.08/Hour	0.00	28.00	0.00	\$0.00
	Discounted Instance Hour \$0.05/Hour	0.00	0.00	0.00	\$0.00
	Backend Instance Hours \$0.08/Hour	0.00	9.00	0.00	\$0.00
	Datastore Storage \$0.008/GBYTE-day	0.01	1.00	0.00	\$0.00
	Logs Storage \$0.008/GBYTE-day	0.01	1.00	0.00	\$0.00
	Taskqueue Storage \$0.008/GBYTE-day	0.00	0.49	0.00	\$0.00
	Blobstore Storage \$0.0043/GBYTE-day	0.00	5.00	0.00	\$0.00

Software as a Service (SaaS)

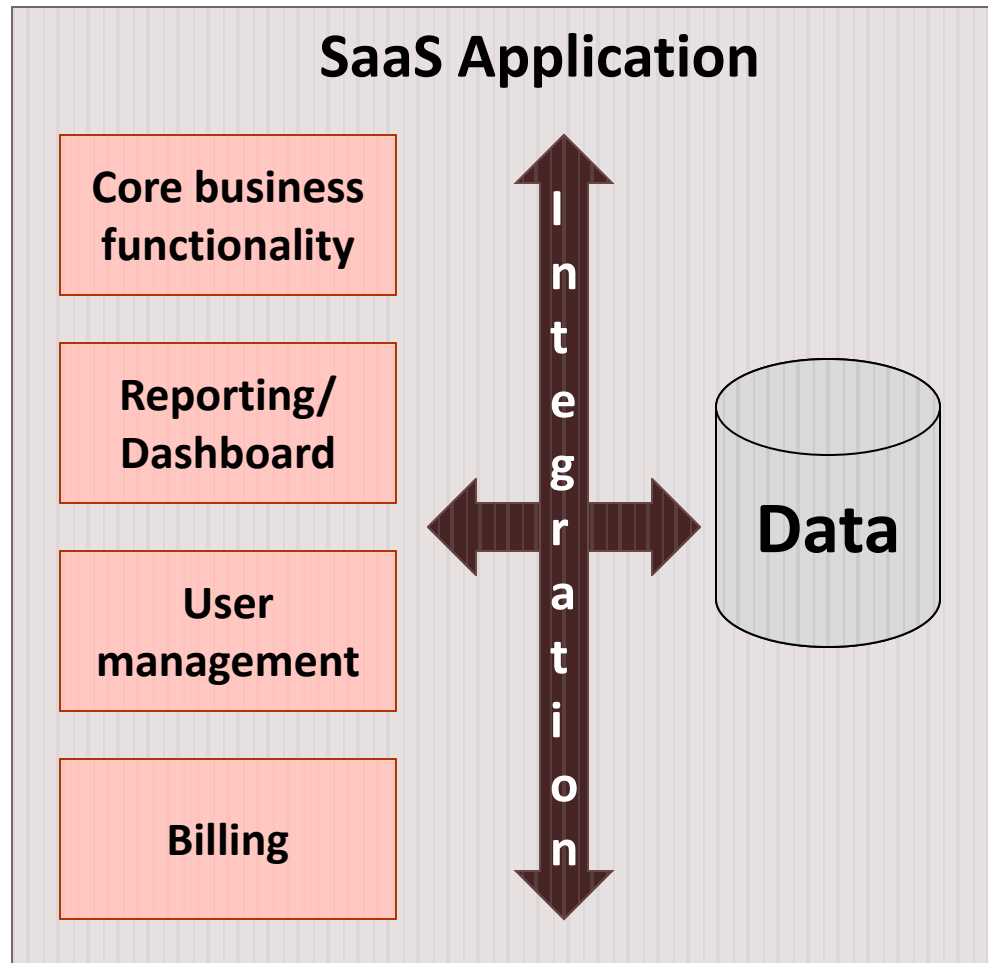
- NIST* definition describes it as:
 - “The **capability** provided to the consumer is **to use the provider’s applications running on a cloud infrastructure**. The applications are **accessible from various client devices** through a thin client interface such as a web browser (e.g., web-based email)”

**NIST = National Institute of Standards and Technology*
Computer Science and Engineering, IIT Ropar

SaaS Examples

- Google Sites
 - Customizable websites, e.g. can add gadgets
 - Integrated with Google services
- rSmart Sakai Learning Management System
 - A feature rich LMS
 - Online class interactions or projects collaborations
- Youtube video streaming
 - Create and manage video channels
 - Edit/enhance videos
 - Analytics

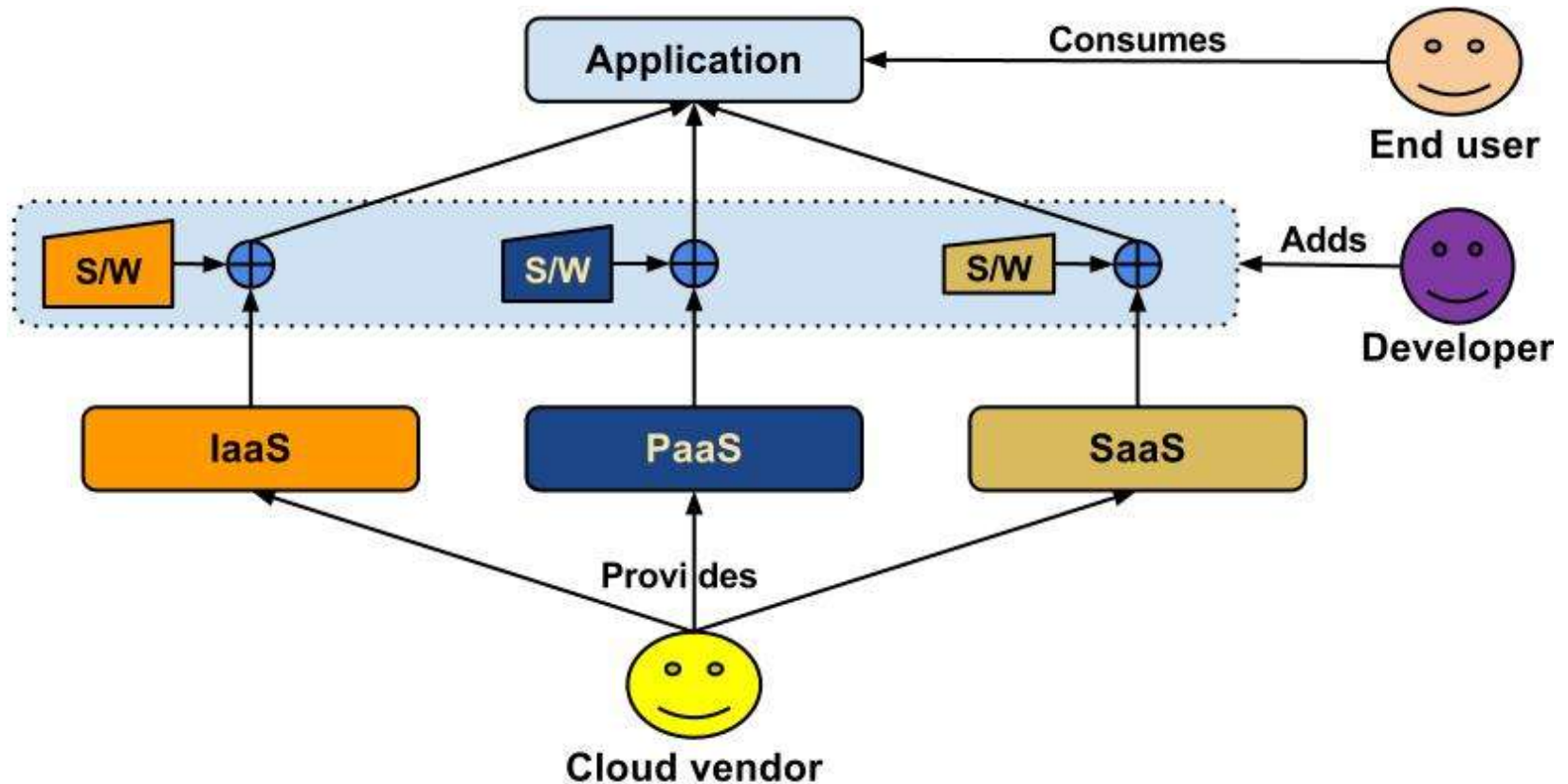
SaaS Architecture



SaaS Cloud Characteristics

- No control of underlying infrastructure
 - Network, servers, operating systems, storage, or individual application capabilities
- Allows control of a limited set of user-specific application configuration settings
- Typically no programming is needed
- User generated data can be exploited by cloud provider
 - Privacy is often an issue

Dependency View for XaaS



Deployment model based

Classification of Cloud Platforms

Public Cloud

- The cloud services are offered to general public for a subscription
 - Services are easy and fast to obtain
- Leverages economies of scale
 - Lowers the costs for consumers
 - Optimization of services due to dedicated focus

Public Cloud

- Why would you use public cloud services:
 - Mainly because it lets you focus on delivering differentiating business value
 - Hardware and other infrastructure managed by cloud provider → **Reduces cost and effort**
 - On-demand processing power, storage, etc. → **Easy scalability**
 - Self service eliminates procurement overheads → **Improved time-to-market for customers**
 - Pay as you go → **No wasted resources because you pay for what you use**

Public Cloud Vendors

- Amazon Web Services
 - Core infrastructural services
 - EC2 for compute, S3 and EBS for storage, Route 53 for networking etc.
 - Specialized services
 - Databases (RDS, SimpleDB etc.), CloudWatch for monitoring, SQS for queuing etc.
- Rackspace
 - Core infrastructural services
 - Service for compute, storage and networking
 - Specialized services
 - Monitoring, load balancers etc.

Public Cloud Vendors

- Google
 - App Engine (PaaS offering)
 - Apps (SaaS offering)
 - Storage and compute
- MS Azure
 - Virtual Machine (IaaS offering)
 - Cloud services (PaaS offering)
 - Online services (SaaS offering)

Private Cloud

- Operated solely for one organization
- Full ownership, control and custody of applications, data and computing assets
- Allows custom configurations of cloud infrastructure
- Often has a homogeneous virtualization environment

Toolkits/Frameworks for Private Clouds

- IaaS cloud
 - OpenNebula
 - Eucalyptus
 - Apache CloudStack
 - OpenStack
 - Nimbus
- PaaS cloud
 - AppScale
 - CloudFoundry Micro

Community Cloud (CC)

- A collaborative effort for sharing infrastructure among several organizations
- These organizations typically form a community
 - E.g. academic institutions, research labs
- Community has common concerns
 - E.g. security, compliance and jurisdiction etc.
- Can be managed/hosted internally by members or by some 3rd party
- A paradigm for Cloud Computing in the community
 - Without dependence on Cloud vendors, such as Google, Amazon, or Microsoft

Hybrid Cloud

- A composition of two or more clouds
 - E.g. private, community or public
- Member clouds are bound together but remain unique entities
 - Allow benefits of multiple deployment models
- Both in-house resources and off-site server-based cloud infrastructure are needed
- Lack flexibility, security and certainty of in-house applications
- Offers flexibility of on-premise applications with fault tolerance and scalability of a public cloud

Motivation and Use Case

- Mainly used to augment the in-house computing
 - For instance, when handling spikes in application load
- Offloading special purpose computing tasks to a cloud based service provider
 - For instance, a cloud based MapReduce cluster

Some Challenges

- Starting and stopping cloud based services/nodes
 - Need to be able to do it on demand
 - For example, how to decide when to start/stop a VM on a cloud?
- Ensuring a homogenous environment
 - How to allow in-house and cloud nodes to discover each other?
- Network issues
 - How to ensure bi-directional communications?
- Inter-cloud communication latencies