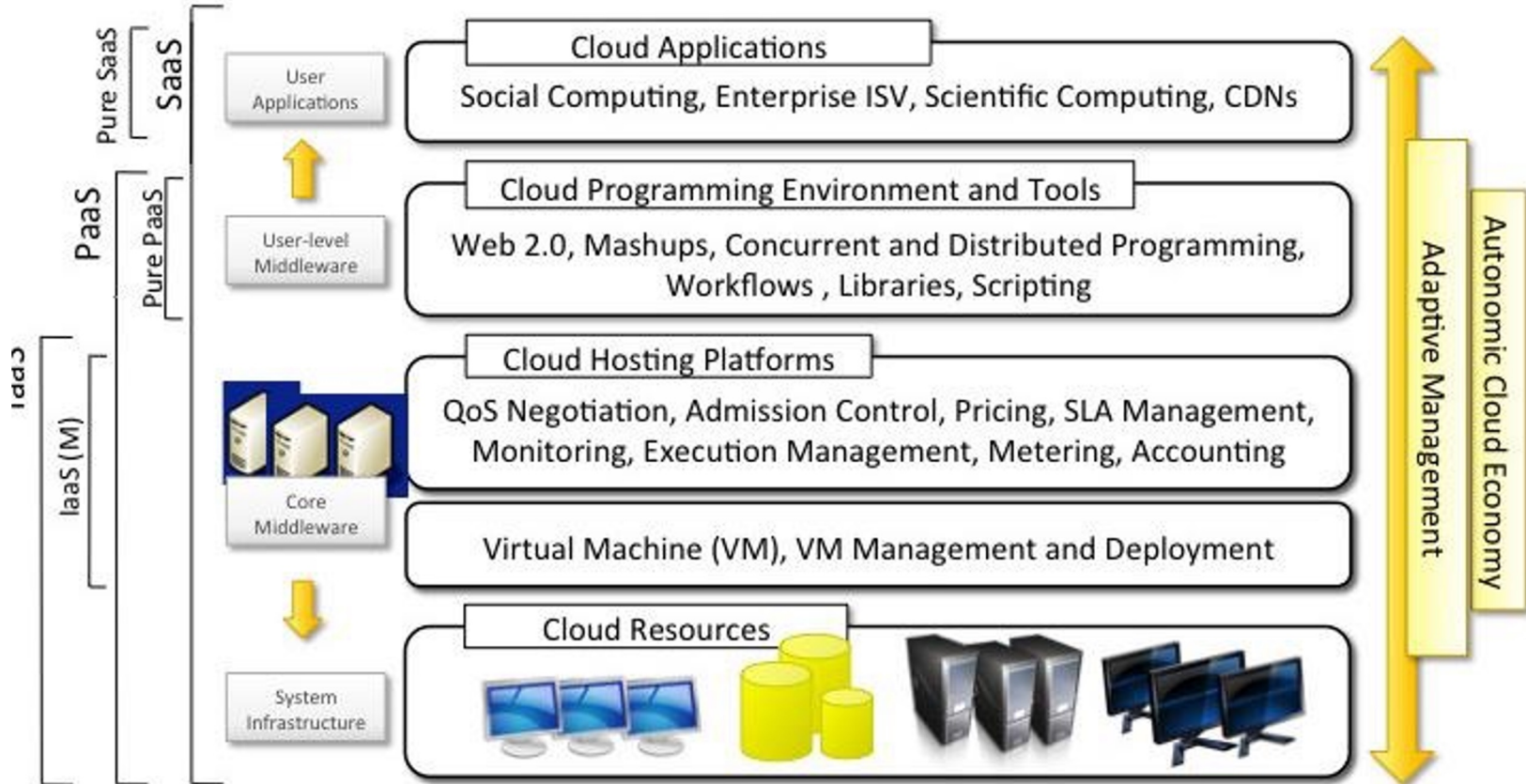


Rashmi Kansakar

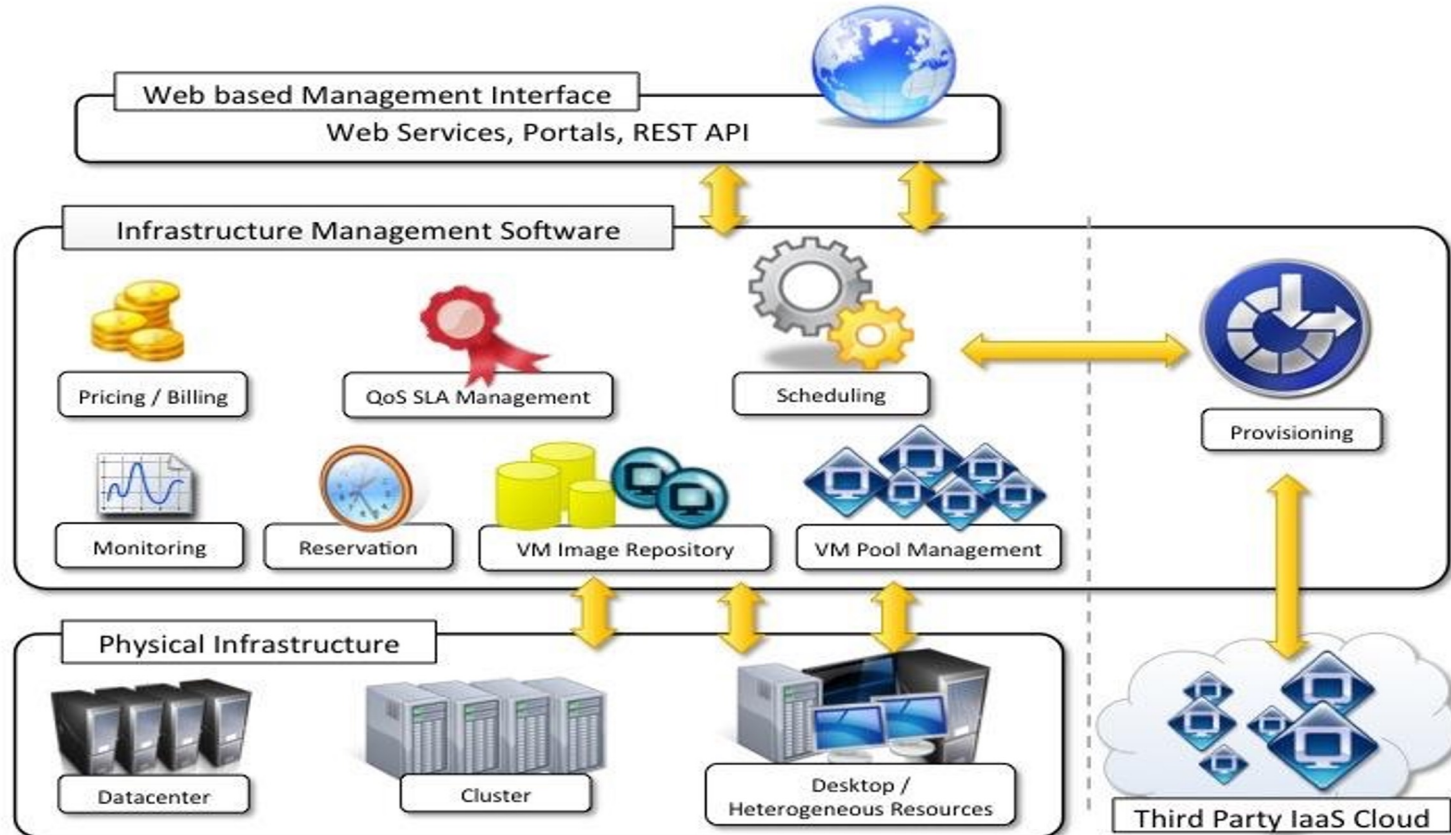
Layered View



Categories

- IaaS - Infrastructure as a Service (AWS)
 - IaaS(M) - Only management layer
- PaaS - Platform as a Service (AppEngine) Use APIs provided to build
 - Pure PaaS - Only middleware, bring hardware
- SaaS - Software as a Service (Gmail)
 - Adaptively scaling to demand

IaaS (*HaaS*)

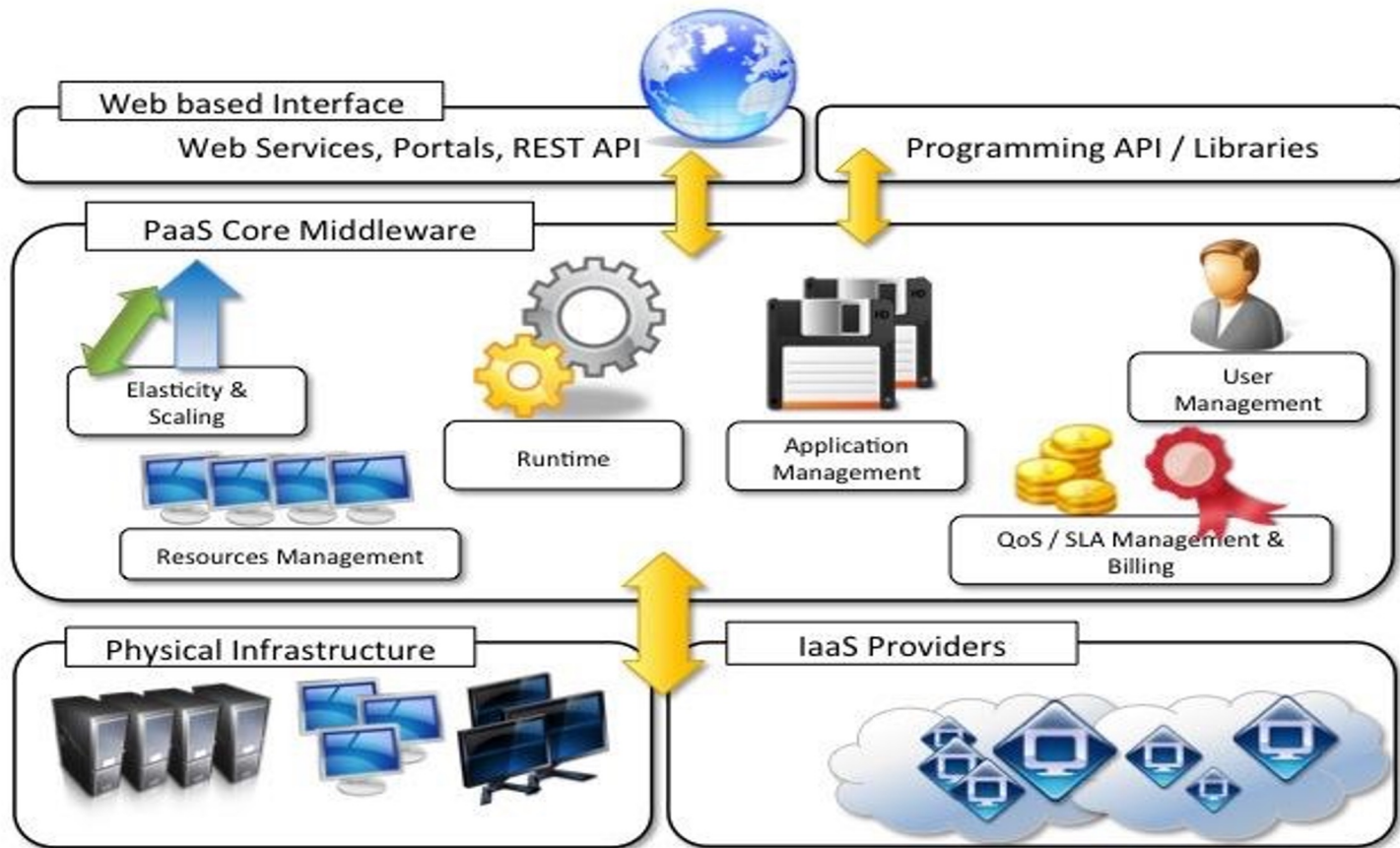


IaaS (*HaaS*)...

- Customers are provided with virtualized hardware and storage on top of which they can build their infrastructure
 - Ex: Amazon EC2 & S3
 - DataPipe
 - Microsoft Azure VMs
 - Samsung Joyent, etc.... (many)

- IaaS(M) - manage cloud infrastructure
 - Ex: Enomaly
 - Eucalyptus
 - OpenNebula
 - OpenStack
 - CloudStack

PaaS



PaaS...

- Application Management!
 - No ability/need to manage infrastructure
- Support Java, .NET, Python, Ruby, etc..
 - Some restrictions for scalability (no write fs)
- Some provide a local dev environment
- Subcategories:
 - PaaS-I : Rapid prototyping - Web 2.0 dev - Ex: Force.com, Longjump
 - PaaS-II : Runtime environment for web - upload code, use their APIs - Ex: Google AppEngine, AppScale, Heroku, Engine Yard

- Subcategories (cont)
 - PaaS III - General purpose apps - Ex: MS Windows Azure (hw as well), Aneka, Apprenda SaaSGrid, Cloud IQ

- Characteristics of PaaS:
 - Runtime Framework - software stack
 - Abstraction - manage software, not HW
 - Automation - Auto scale resources
 - Cloud Services - API Ex: Storage, identity management, computation

➤ Issues....

- Vendor lock-in
 - Runtime environment: Force.com
 - AppEngine: standard, Python, PHP, Java...
 - APIs may be proprietary

➤ Cost benefit

- Reduces management costs for consumer
- Vendor does hardware/software updates -> saves them money

➤ Fueling startup culture!

- Don't need to be an infrastructure expert

SaaS

- Access to software from web browser
 - No software to install, no upfront costs
- Good for 'one size fits all' systems - little customization needed
- Good fits:
 - Customer relationship management (CRM)
 - Enterprise resource planning (ERP)
- Pay-as-you-go (PAYG) pricing

- Multi-tenancy - multiple customers - same system
 - Optimized resources
 - Continuous upgrades
 - Distribute costs
 - Developers know how to run the best
 - Significant cost savings - no distribution, little support needed
 - Rapid implementation

SaaS...

- Very successful delivery model!
- SaaS 2.0 - Integrate reliable services
 - Use SLAs
 - Adds another dimension & options
 - Could promote open market
- Plugins becoming popular - extend
 - Facebook, LinkedIn
- Office automation
 - Google Documents, Zoho Office, etc

Types of Clouds (review)

- Public
 - EC2
- Private
 - Customer information protection
 - Infrastructure ensuring SLA
 - Compliance w/ procedures and operations
- Hybrid or heterogeneous
 - Cloudbusting - using public cloud for spikes
- Community [example](#)
 - Media Industry, Healthcare, Energy, Scientific

➤ Why Cloud Computing?

- Reduce capital IT costs
- Eliminate depreciation
- Replace software license with subscription
- Reduce maintenance & admin costs of IT

Open Challenges

- Defining Cloud Computing!!!!
- Cloud Interoperability
- Standards
- Scalability & Fault Tolerance
- Security, trust, privacy
- Changing role of IT in a business
 - What does a 'new' IT dept do?
 - Compliance?
 - Loss of control of data?

Summary

- Service models: SaaS, PaaS, IaaS
- Deployment models: Public, Private, Hybrid, Community
- Driving force behind cloud --> \$\$\$\$
- Issues & Challenges