Karlsruhe Institute of Technology

Introduction to Cloud Computing

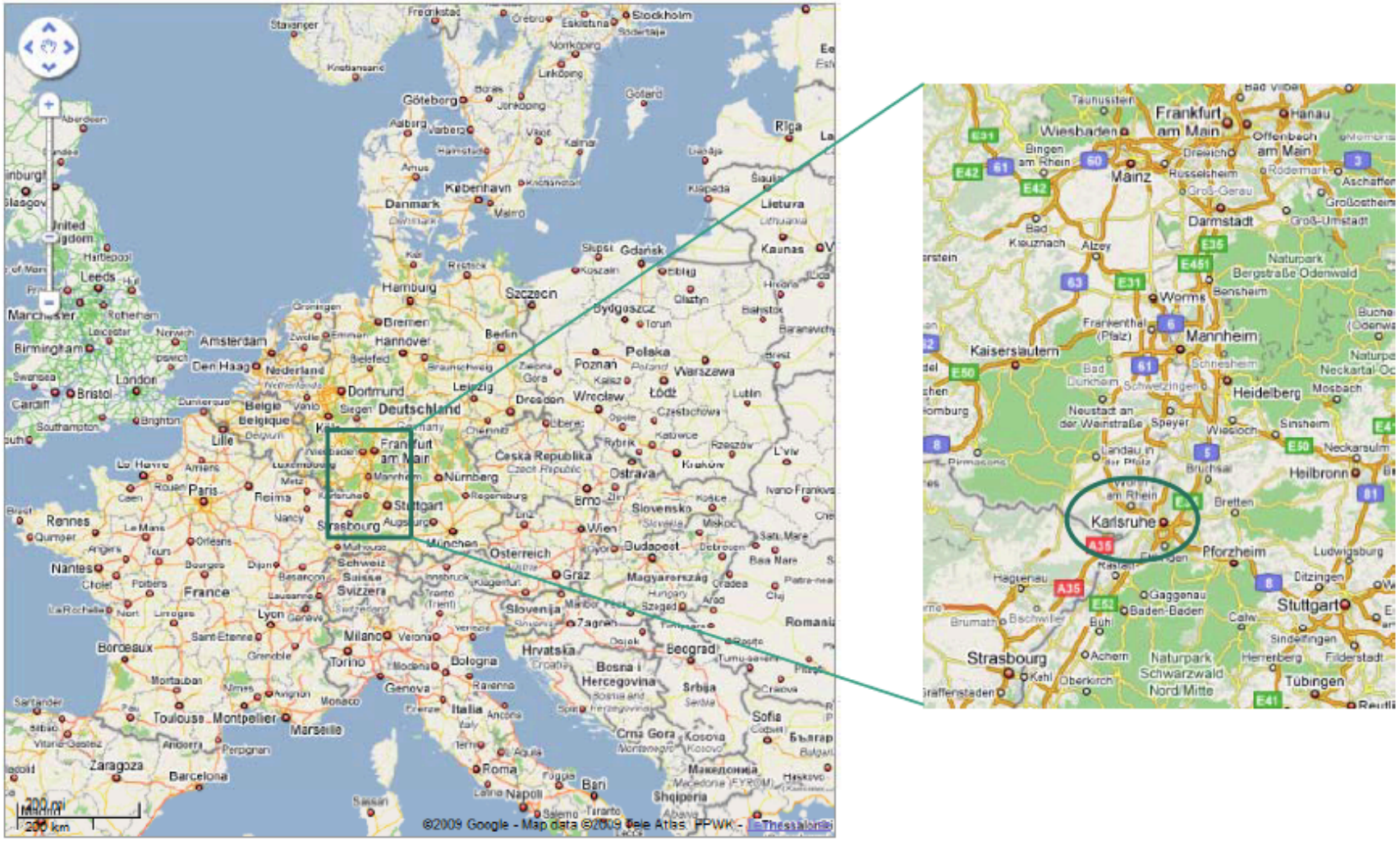
Dr. Marcel Kunze
Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
marcel.kunze@kit.edu

Sousse Summer School 2012, Tunisia

About the Speaker: Marcel Kunze

- Karlsruhe Institute of Technology (KIT), Germany
  - Head of Research Group Cloud Computing
  - Engineering Mathematics and Computing Lab (EMCL)
  - Steinbuch Center for Computing (SCC)
- From 2002-2009:
  - Department Leader “Integration and Virtualization”
  - Department Leader “Grid Computing and e-Science”
  - Forschungszentrum Karlsruhe
- Prior to 2002:
  - Stanford Linear Accelerator Center
  - Ruhr University Bochum, Germany
  - Karlsruhe University, Germany

Karlsruhe, Germany

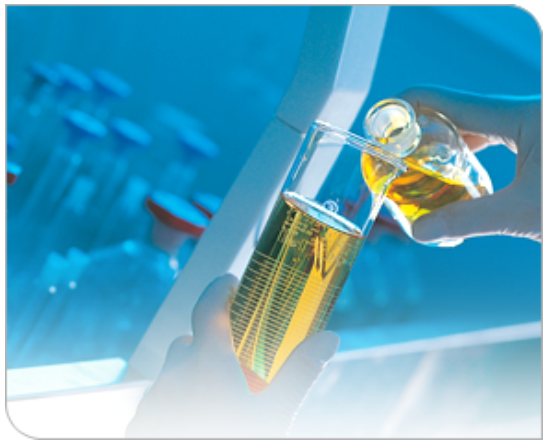


Karlsruhe Institute of Technology (KIT)

Largest European scientific institution

Main topics: Energy, Nanotechnology, Astrophysics, Engineering

Mission:



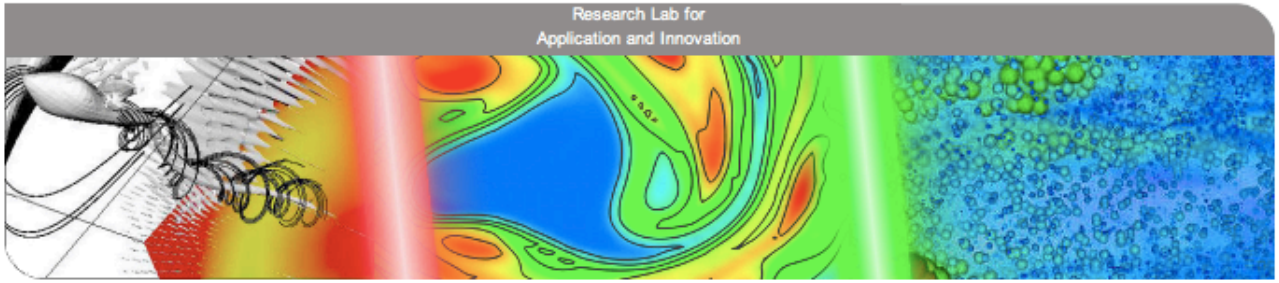
- Research

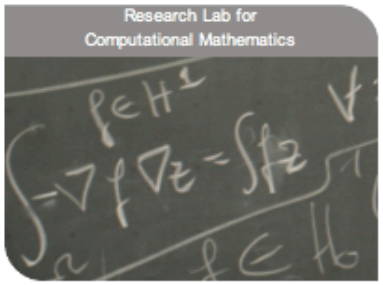
- Education

- Innovation

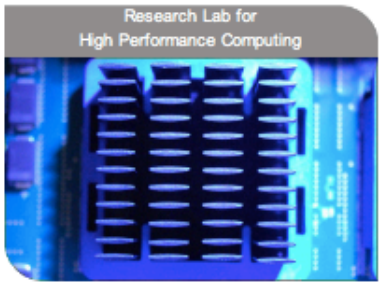
www.kit.edu

Engineering Mathematics and Computing Lab





EMCL



http://www.emcl.kit.edu/

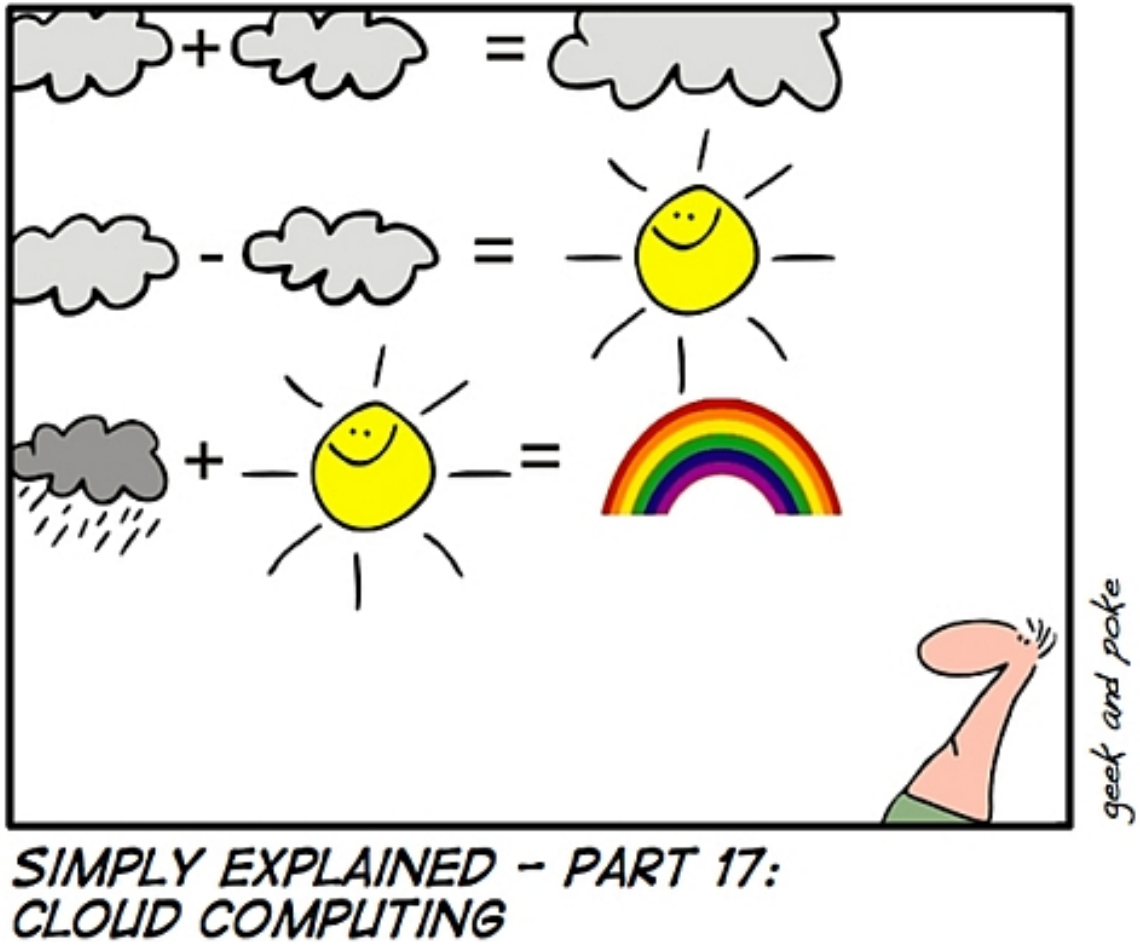
- **Interdisciplinary research focusing on innovation**
- **Scientific Computing, numerical optimization, simulation, High Performance Computing**

Cloud Computing Roadmap

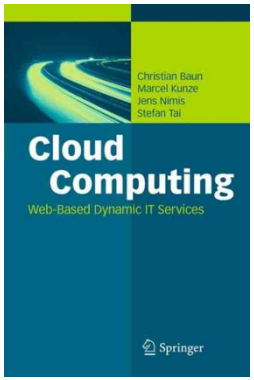
1. Fundamentals
2. Amazon Web Services
3. Cloud Management
4. Cloud Architecture
5. Programming Models
6. Applications

1. Fundamentals

What is Cloud Computing ?



Cloud Computing: Definition



"Building on compute and storage **virtualization**, ***cloud computing*** provides **scalable**, **network-centric**, abstracted IT infrastructure, platforms, and applications as **on-demand** ***services*** that are billed by consumption."

C.Baun, M.Kunze, J.Nimis, S.Tai: Cloud Computing, Informatik im Fokus, Springer 2009-2011

What are Services?

Business View

Service =
Deliverable, vs. products

- Interactivity
- Individuality
- Immateriality

Comprehensive View

Service =
Competence-based activity to co-create value

IT View

Service =
Web-based software construct

means to „transport" value

SOA, interaction via Internet

Joint value creation and ICT

There are five essential characteristics of cloud services [NIST]

[NIST]: http://csrc.nist.gov/groups/SNS/cloud-computing/

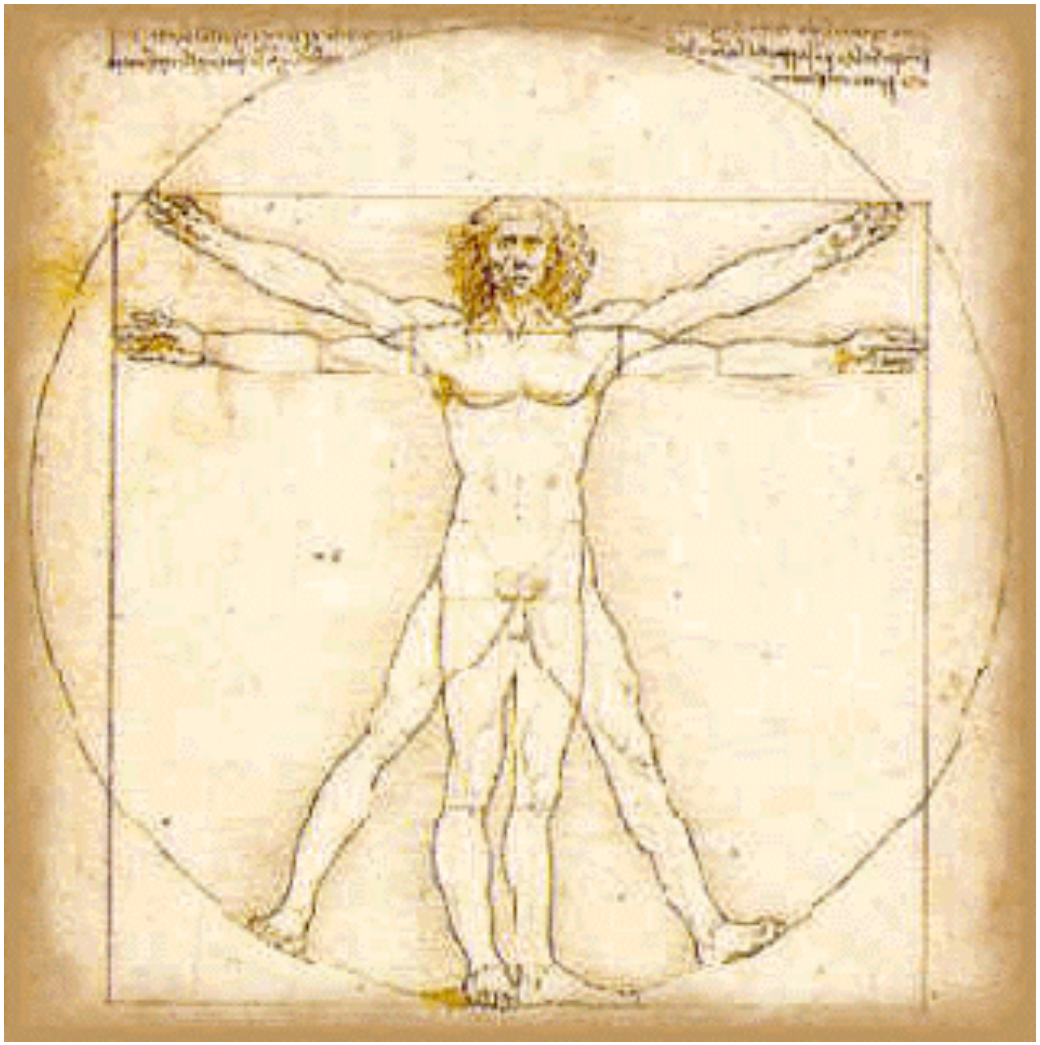
Elastic Scalability

On-demand Self-Service

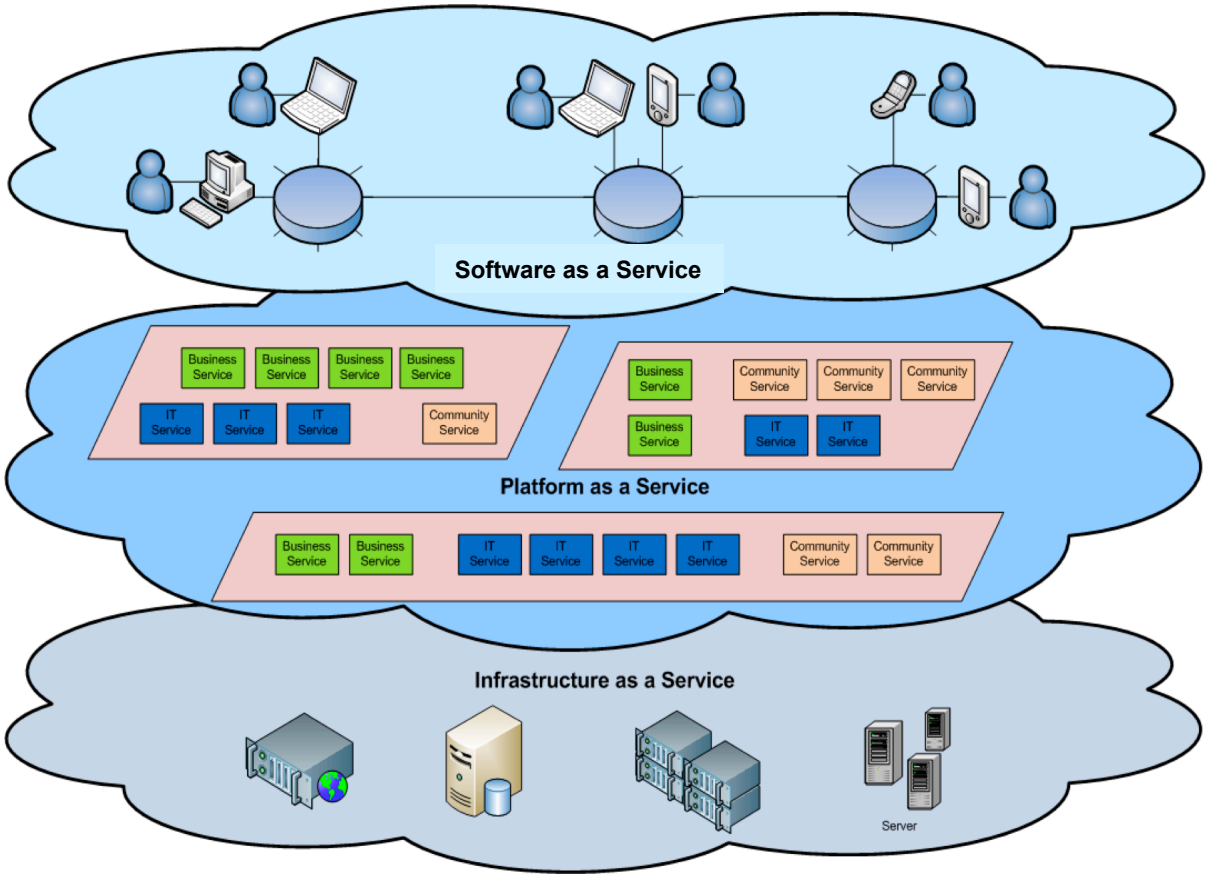
Ubiquitous Network Access

Resource Pooling

Measured Service



Service Delivery Models



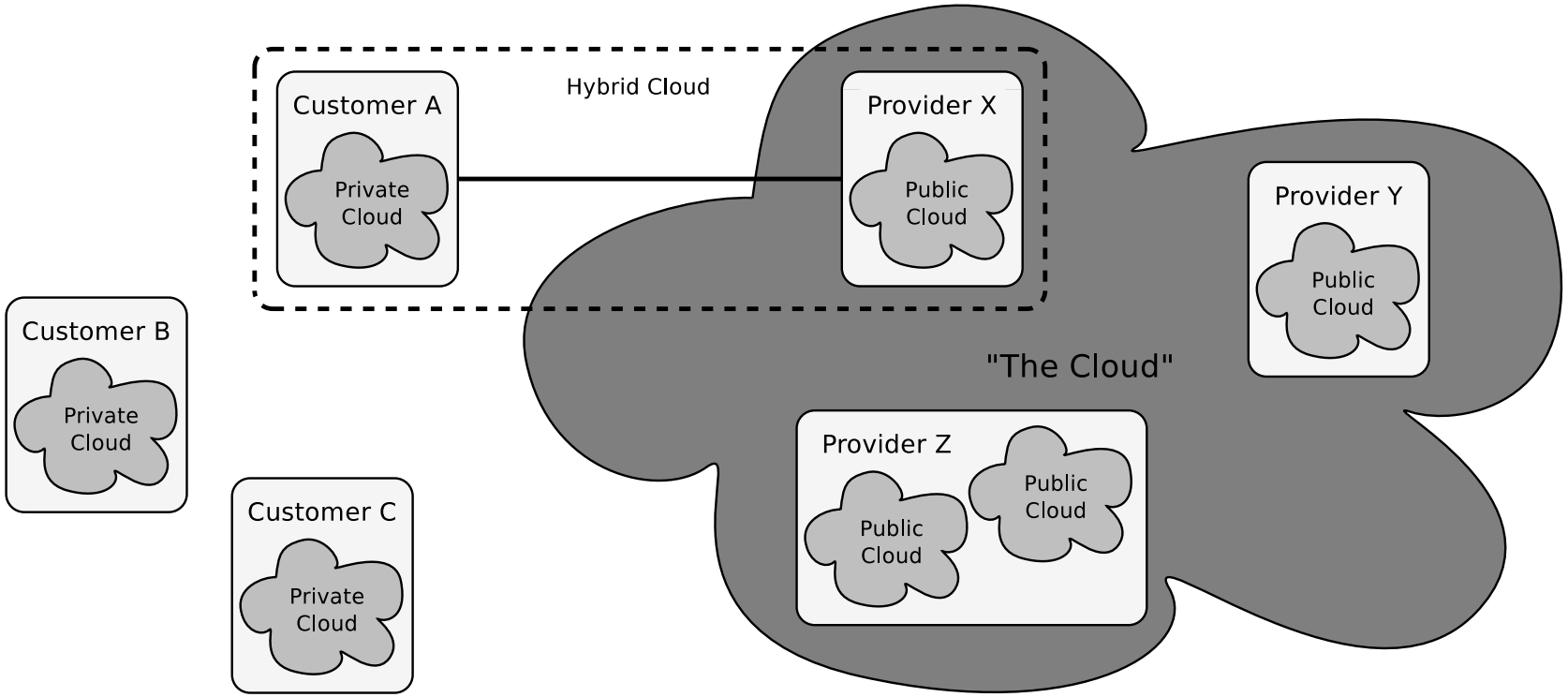
SaaS
Web Applications like Google Apps, Salesforce etc.

PaaS
Development or execution environments like e.g. Google App Engine, Windows Azure

IaaS
Offerings like Google Storage, Amazon Web Services etc.

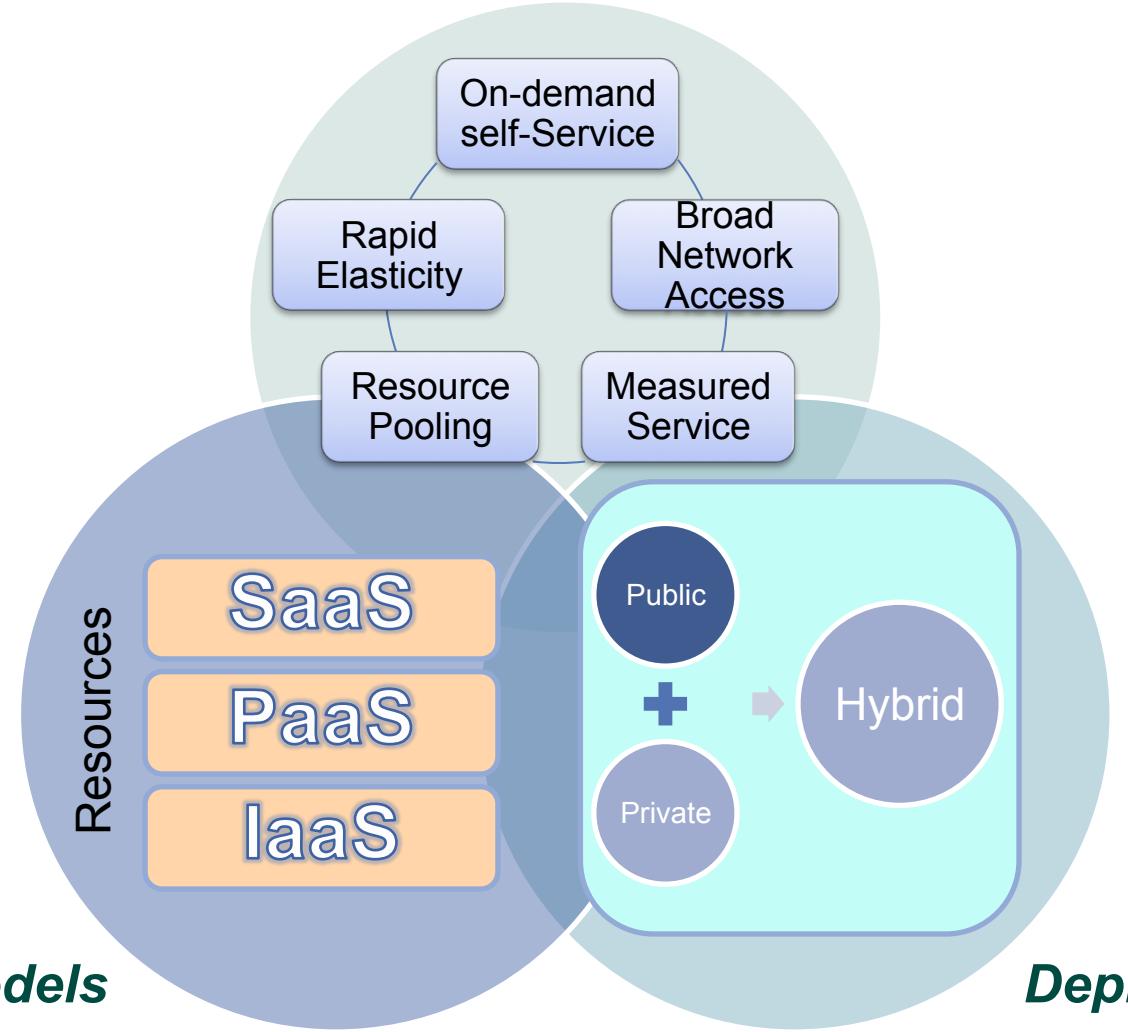
- **Boost all kinds of applications in enterprise and in science**
  - **Legacy applications**
  - **New cloud applications with advanced features**

Service Deployment Models



- **Private cloud: Customer and provider belong to the same organization**
- **Public cloud: Customer and provider belong to different organizations**
- **Hybrid cloud: Combination of private and public cloud**
- **Community cloud: Stakeholders share resources to achieve common goal**

Cloud Computing: Conceptional View [NIST]



What are Resources ?

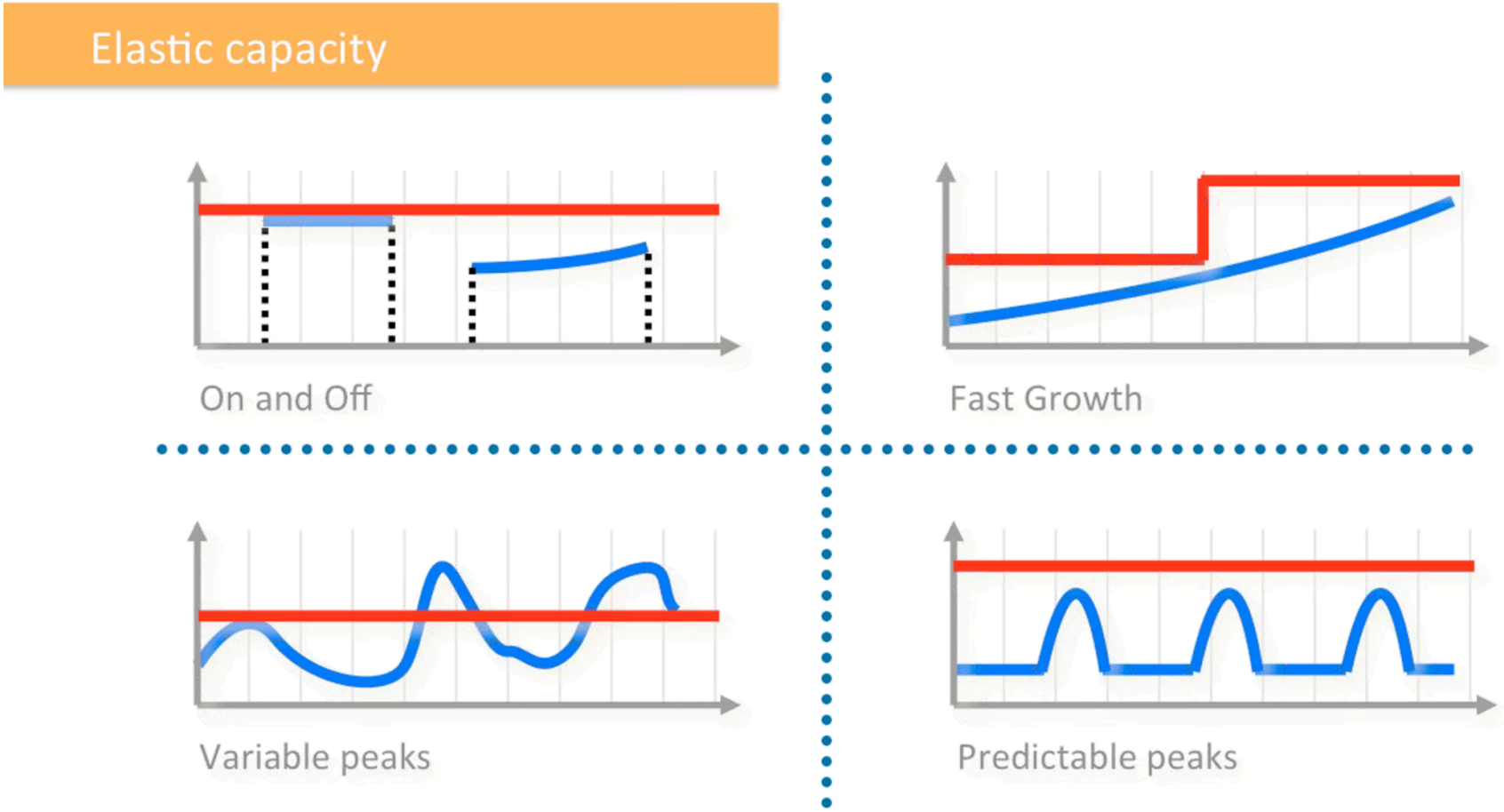
- In general
  - Computing
  - Data
  - Network
  - Humans (Social networks)

- In scientific environments this translates to
  - High Performance Computing (HPC)
  - High Throughput Computing (HTC)
  - High Speed Networking
  - Virtual Organizations (VO)

Problem: Efficient Use of Resources

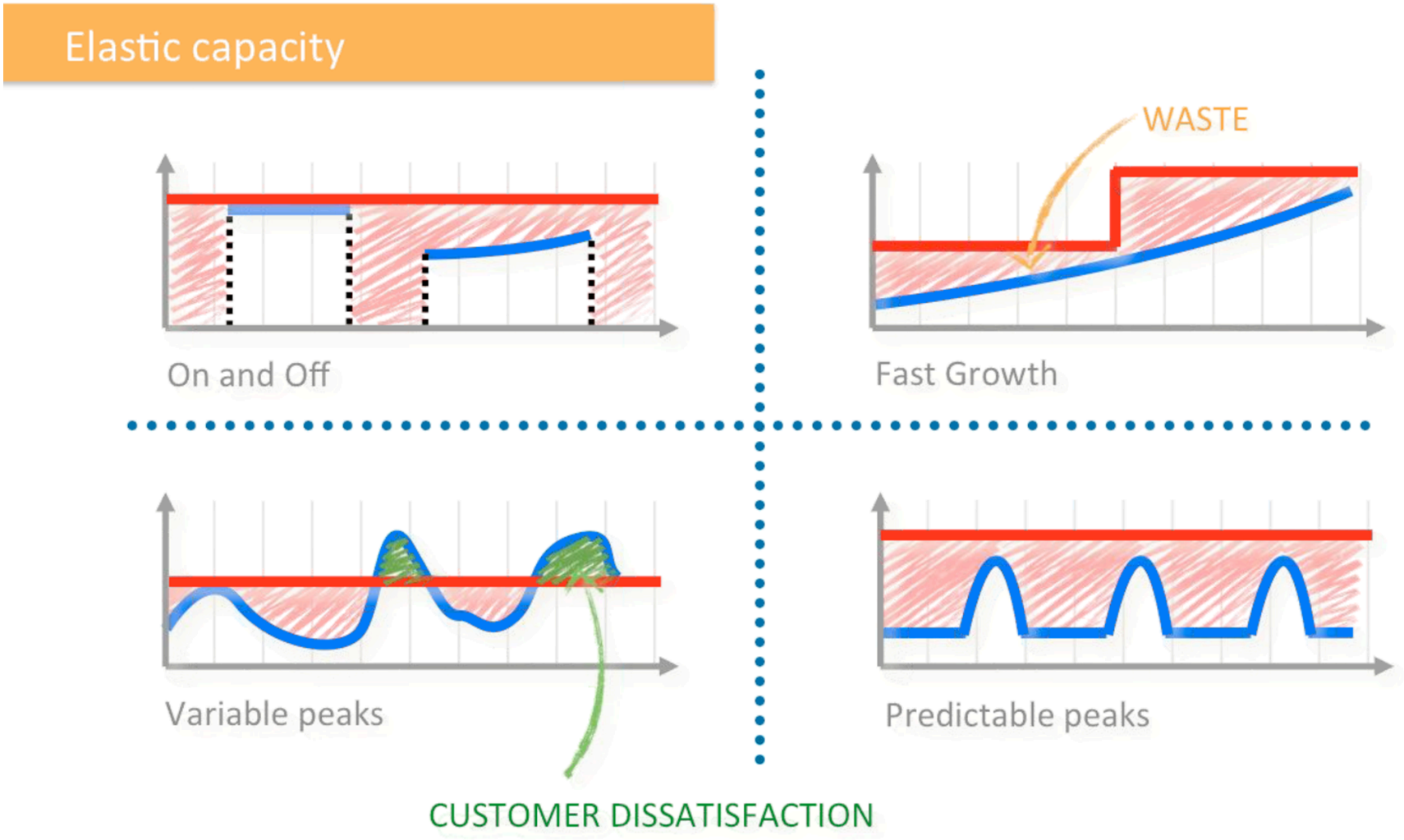
- Concurrent use by various customers (tenants) and scheduling systems
- Varying requirements wrt. operating system, software, hardware, etc.
- Unpredictable load

Resource Demand and Provisioning (1)

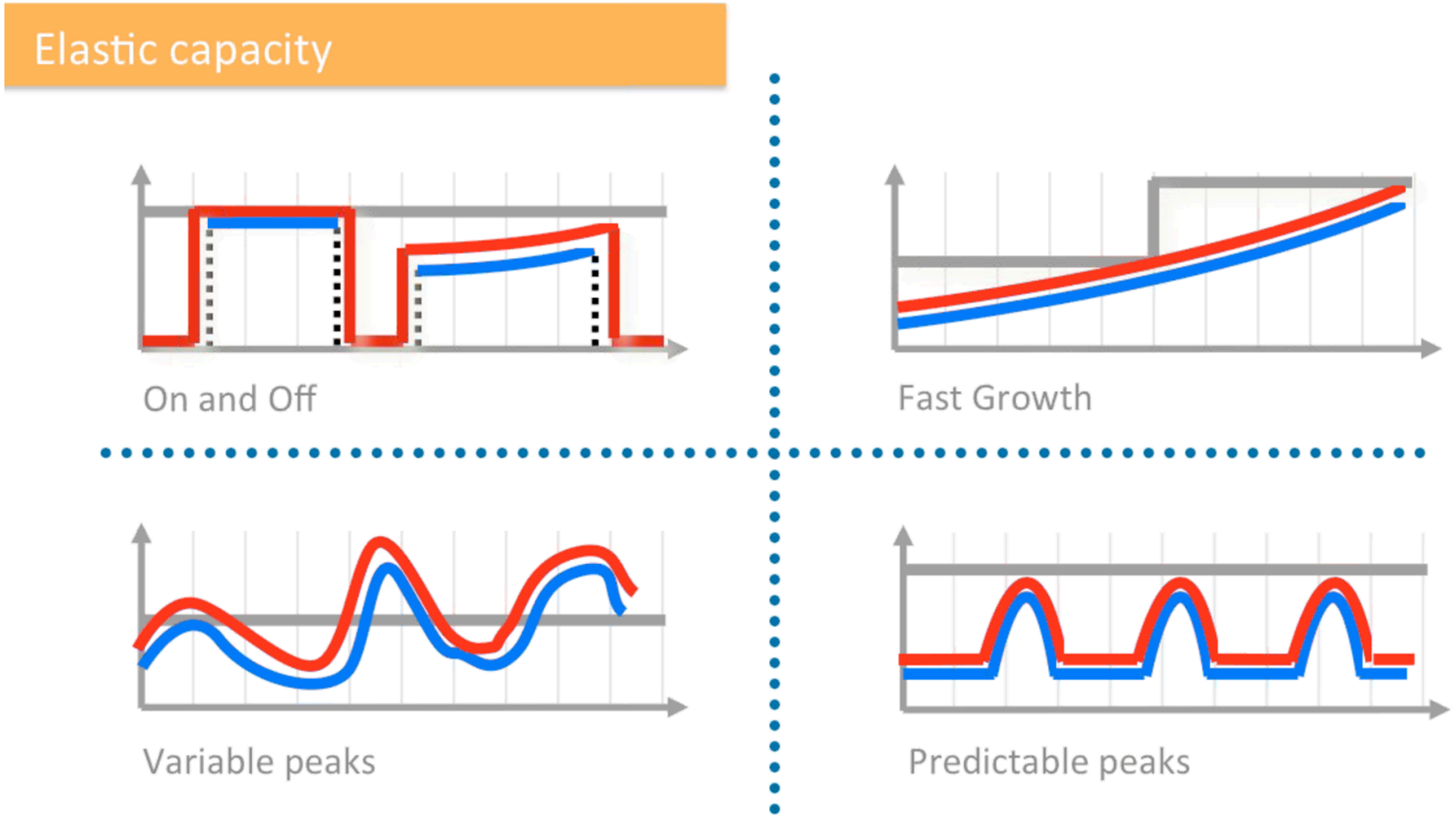


- How accurately can we do the resource planning to follow demand?
- Fixed cost (red) versus variable cost (blue)

Resource Demand and Provisioning (2)



Resource Demand and Provisioning (3)



- **Cloud computing matches resource demand and provisioning**
- **Cloud computing solves the resource provisioning problem**

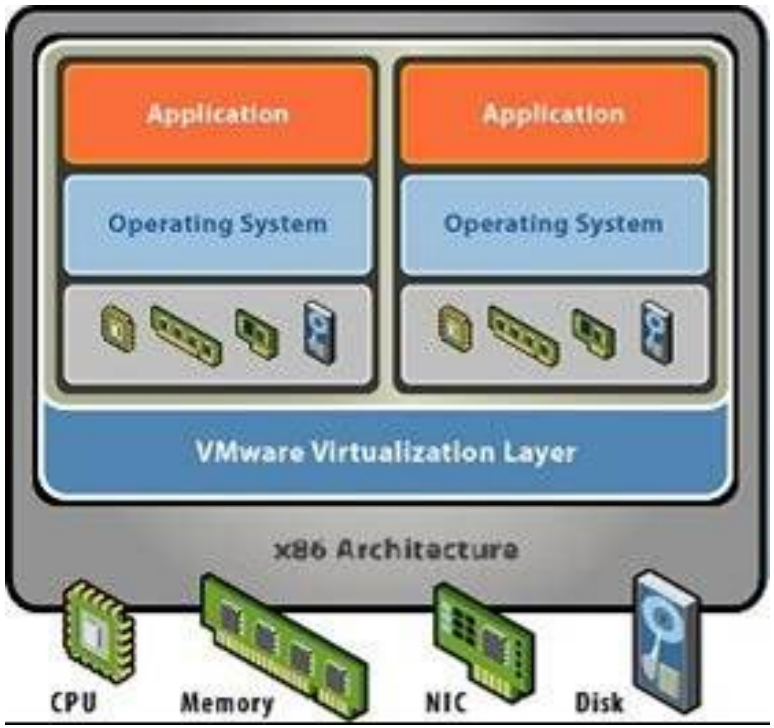
Multi-Tenancy and Partitioning

- **Multi-Tenancy**
  - Isolation of workloads
  - Separation of customers
  - Customers gain administrative privileges
- **Partitioning of Resources**
  - Classical solution: Separation of **servers**
    => Physical Resource Sets (PRS)
  - Cloud solution: Separation of **services**
    => Virtual Resource Sets (VRS)

Virtualization

- **Virtualization is a technique that separates the operating system from the physical computer hardware, and interposes a layer of controlling software (hypervisor) between the hardware and operating system.**
- **Different types of virtualization systems (from Goldberg)**
  - Type 1: hypervisor between “bare metal” and guest operating system
    Examples: VMware, Xen, KVM
  - Type 2: hypervisor between host operating system and guest operating systems
    Examples: Virtual Box, VMware Workstation, Parallels for Mac

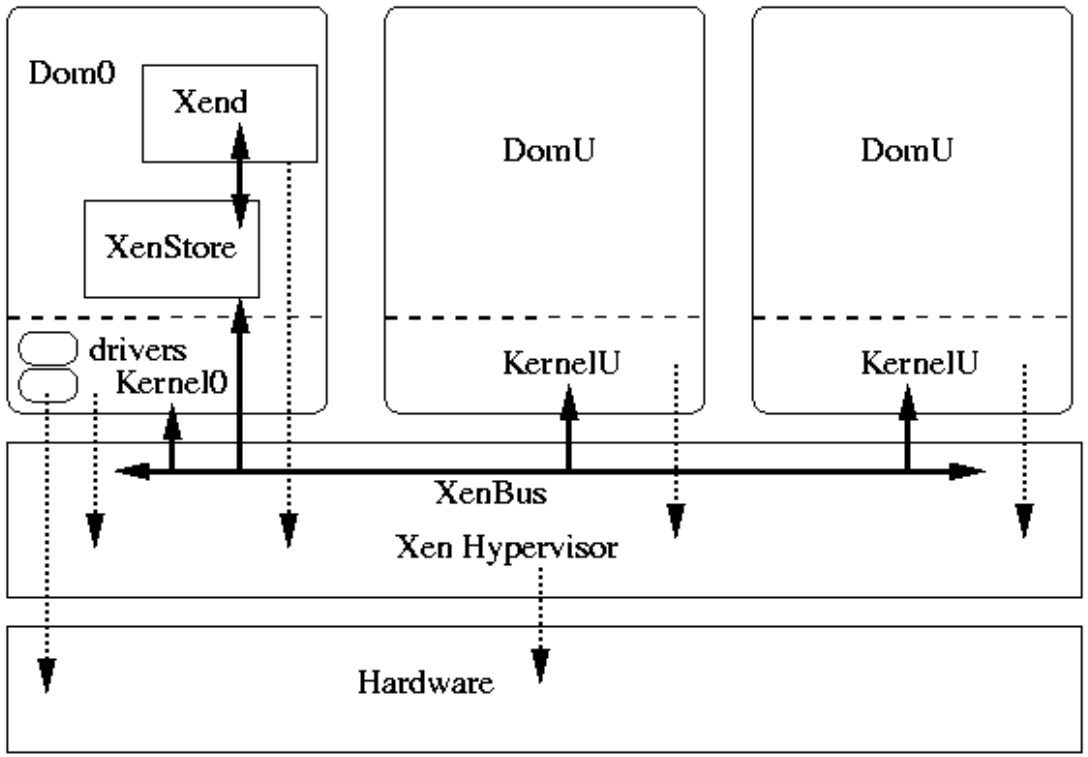
Virtual Machine (VM)



- **Use virtual rather than physical resources**
- **Abstract view rather than physical view on infrastructure**
- **Hypervisor = control program**
- **Partitioning**
  - Resource pooling
  - Multiple OS on single server
- **Isolation**
  - No side effects between VMs
- **Encapsulation**
  - Store VMs together with VM configuration in a file

Source: VMware

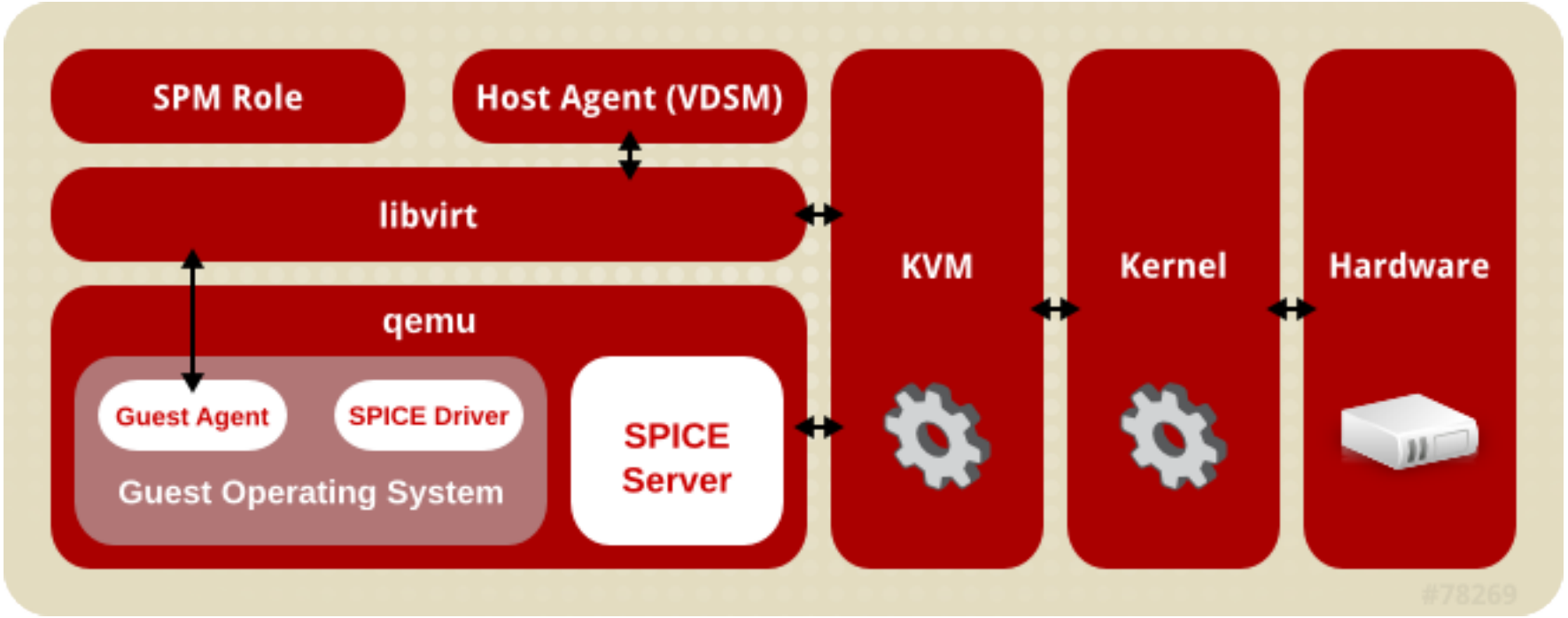
Type 1: Xen



Source: libvirt.org

- **First hypervisor being integrated into the Linux kernel (2003)**
  - Uses *paravirtualization:* Guest OS runs a modified operating system to interact with hypervisor and kernel
  - Host OS runs as Domain0
  - Guest OS runs as DomainU

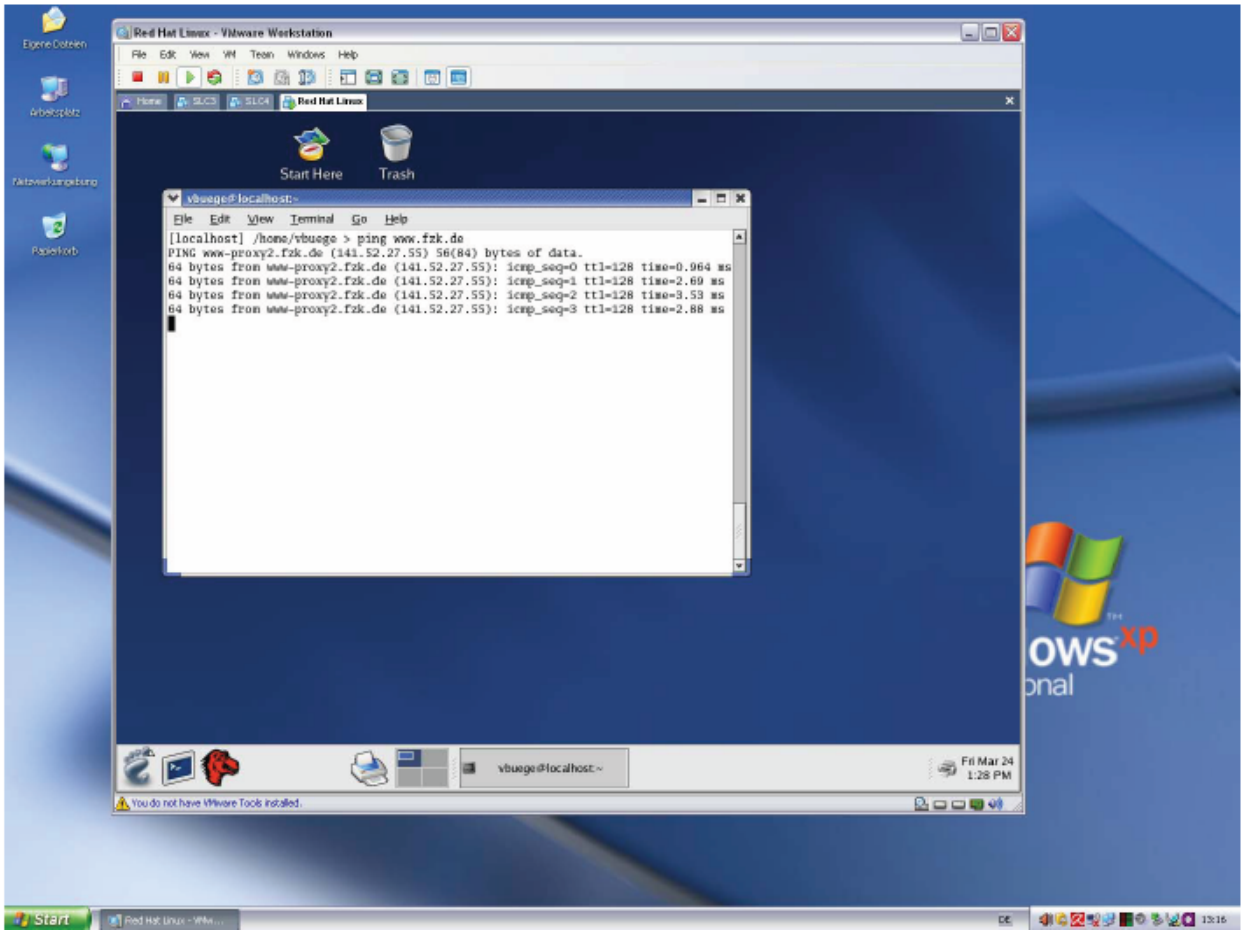
Type 1: Kernel Virtual Machine (KVM)



Source: redhat.com

- **Virtual machine runs as a regular process (Unmodified OS)**
- **Resource management with OS tools**

Type 2: VMware Workstation (Fusion)



- Concurrent use of Linux on Windows, or Windows on Mac OS X
- VMware Player (free): Allows to run virtual machine images

Virtual Desktop (Thin Client, Green PC)

- **Application area**
  - Virtualization of desktop PCs
  - Virtualization of PC Pools
  - Operating system runs in the virtual infrastructure or in the cloud
- **„Look and Feel“ of a normal PC, but: cool and quiet!**
- **Reduced TCO in comparison to unmanaged PC (Factor 3-4); Fraunhofer study:** http://it.umsicht.fraunhofer.de/PCvsTC
- **Improved environmental performance**
- **Central management**
- **Improved security: Data are always stored in the data center**
- **Improved energy balance; Replacement of 1000 PCs reduces power consumption by 50-100 kW!**
- **New: Google ChromeBox/ChomeBook**
  - **Cloud Desktop**

chrome

Virtualization Tools

- **Additional mechanisms in Linux**
  - Libvirt / virtio
    - Library and utilities for virtualization systems interoperability
  - Brctl
    - Linux virtual network bridge control package
  - Cgroups
    - Linux feature for controlling resource use of processes
- **Network virtualization**
    - Virtual LANs (VLANs) are best, but need to be configured in switches
    - **OpenFlow** simplifies network management and makes it scalable. http://www.openflow.org

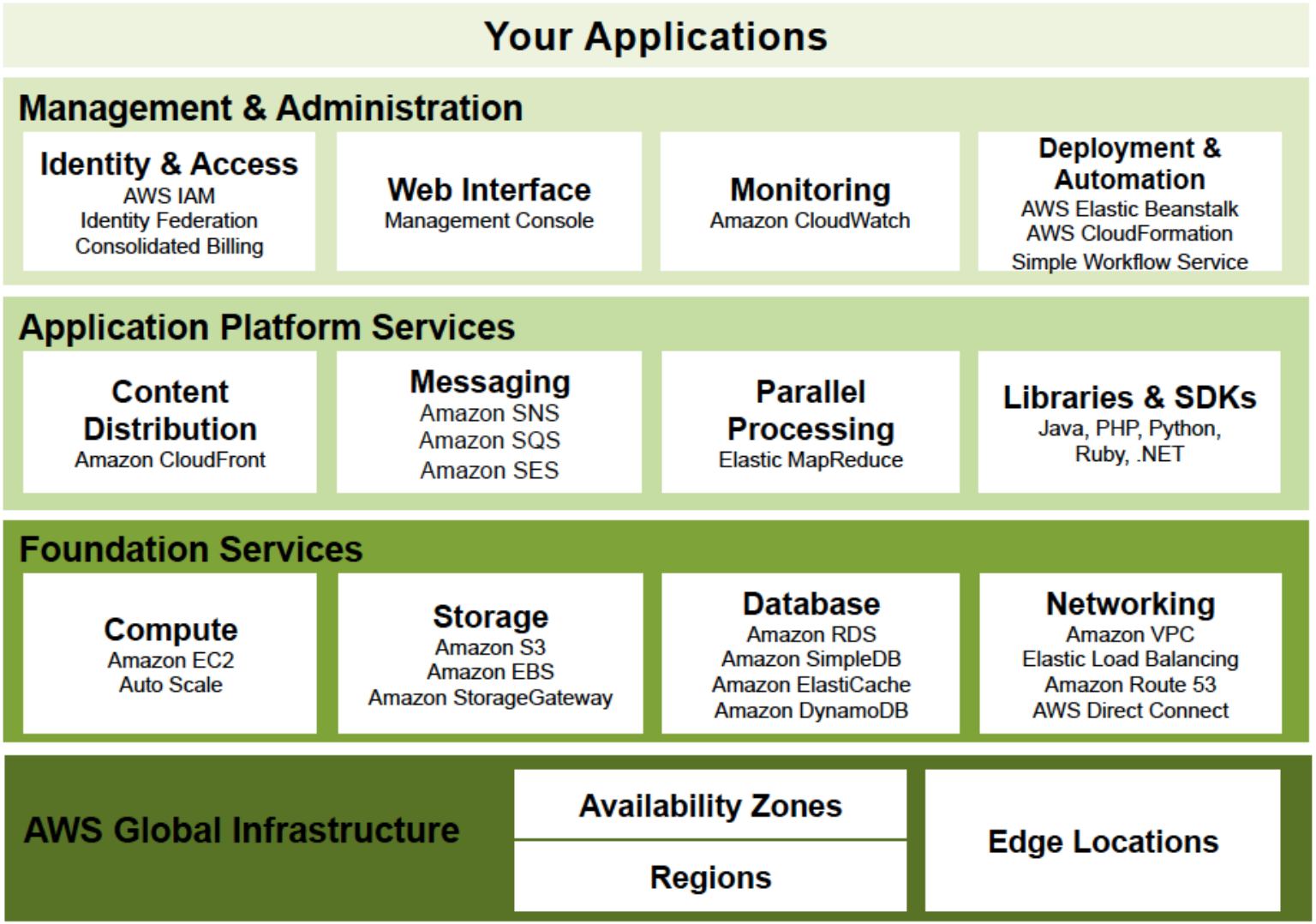
Is Virtualization already Cloud Computing?

- **No, there are some missing aspects:**
  - Business models (Accounting, billing)
  - Self-service
  - Scalability (“Unlimited resources”)
  - Accessibility over Internet (Web services)

- **Virtualization is an enabling technology for Cloud Computing**

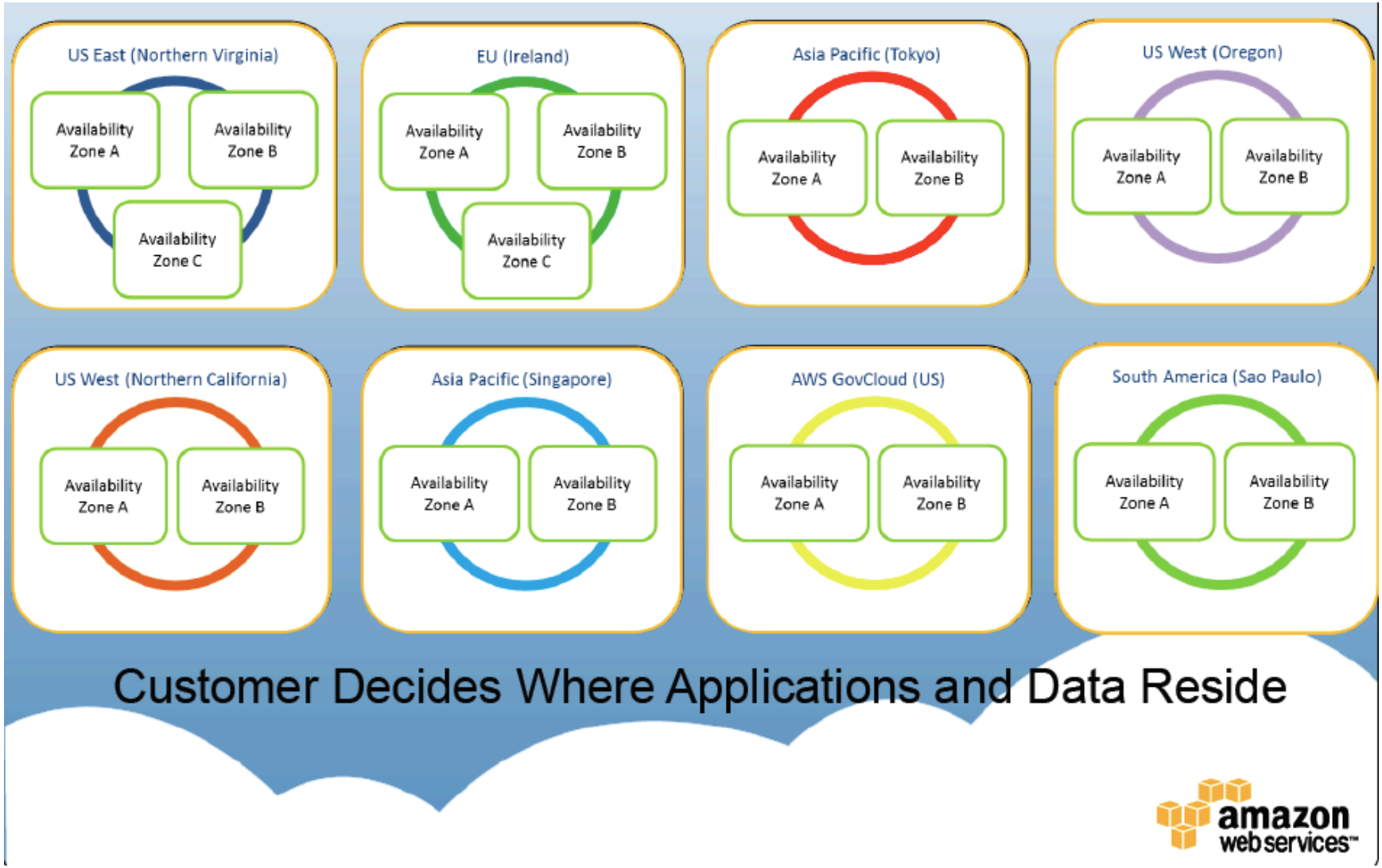
2. Amazon Web Services

Amazon Web Services (AWS)



Source: Amazon

AWS Regions and Availability Zones

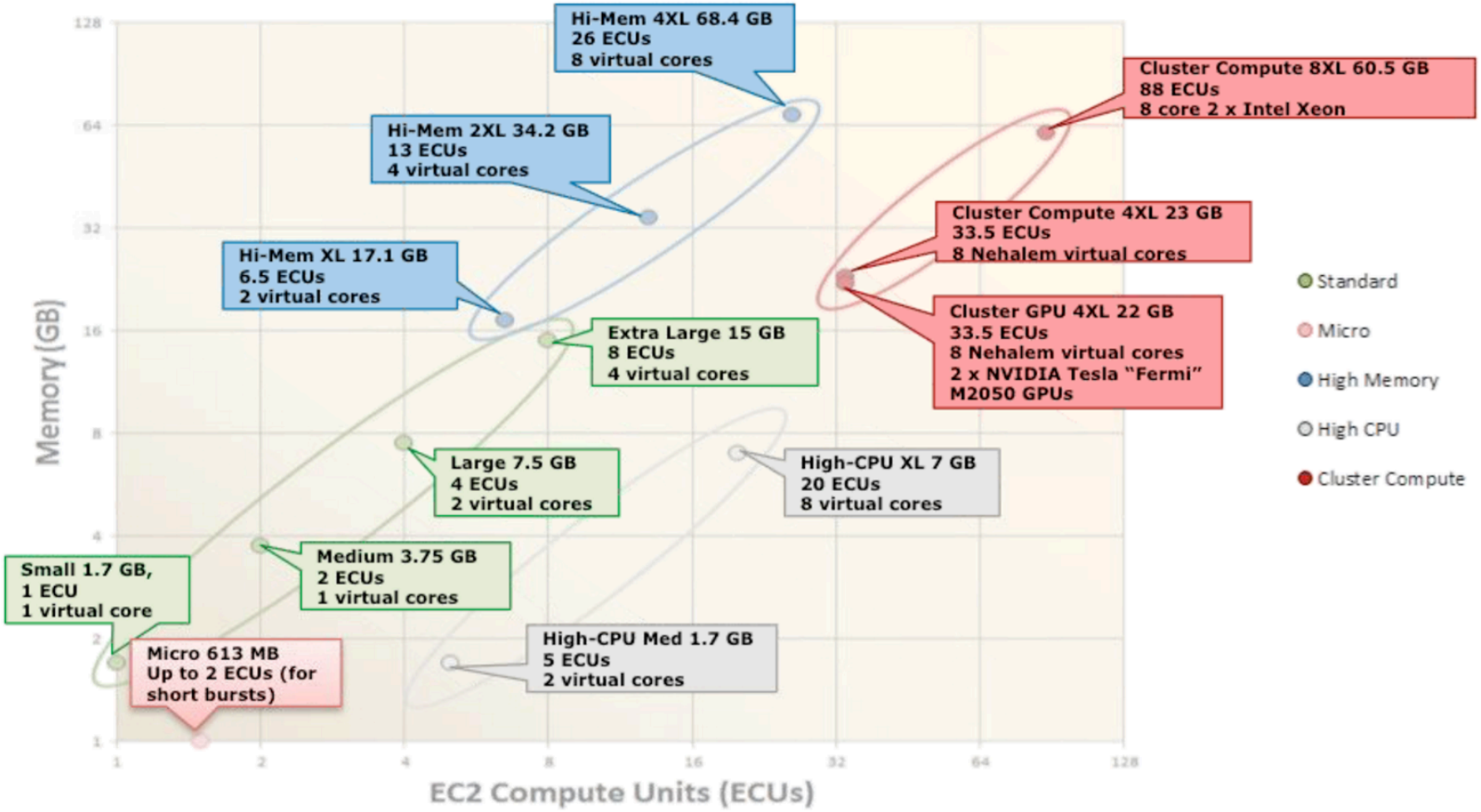


AWS Elastic Compute Cloud (EC2)

- **Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers.**

- **Features**
  - Virtual machines running Windows or Linux
  - Full Windows admin or Linux root privileges
  - Ephemeral storage, Elastic Block Storage and SSDs
  - Instance types ranging from t1.micro to cc2.8xlarge
    - 2 * Intel Xeon ES-2670
    - “Sandy Bridge” Architecture
    - 16 cores w/ HT
    - 60.5 GB RAM
    - 3.4 TB disk
    - HVM
  - High performance instances have 10 Gigabit full bisection networking bandwidth

EC2 Instance Types



Source: aws.amazon.com

EC2 Prizing Models

- **On-Demand**
- **Reserved Instances**
  - Light
  - Moderate
  - Heavy
- **Spot**
- **Dedicated Instances**

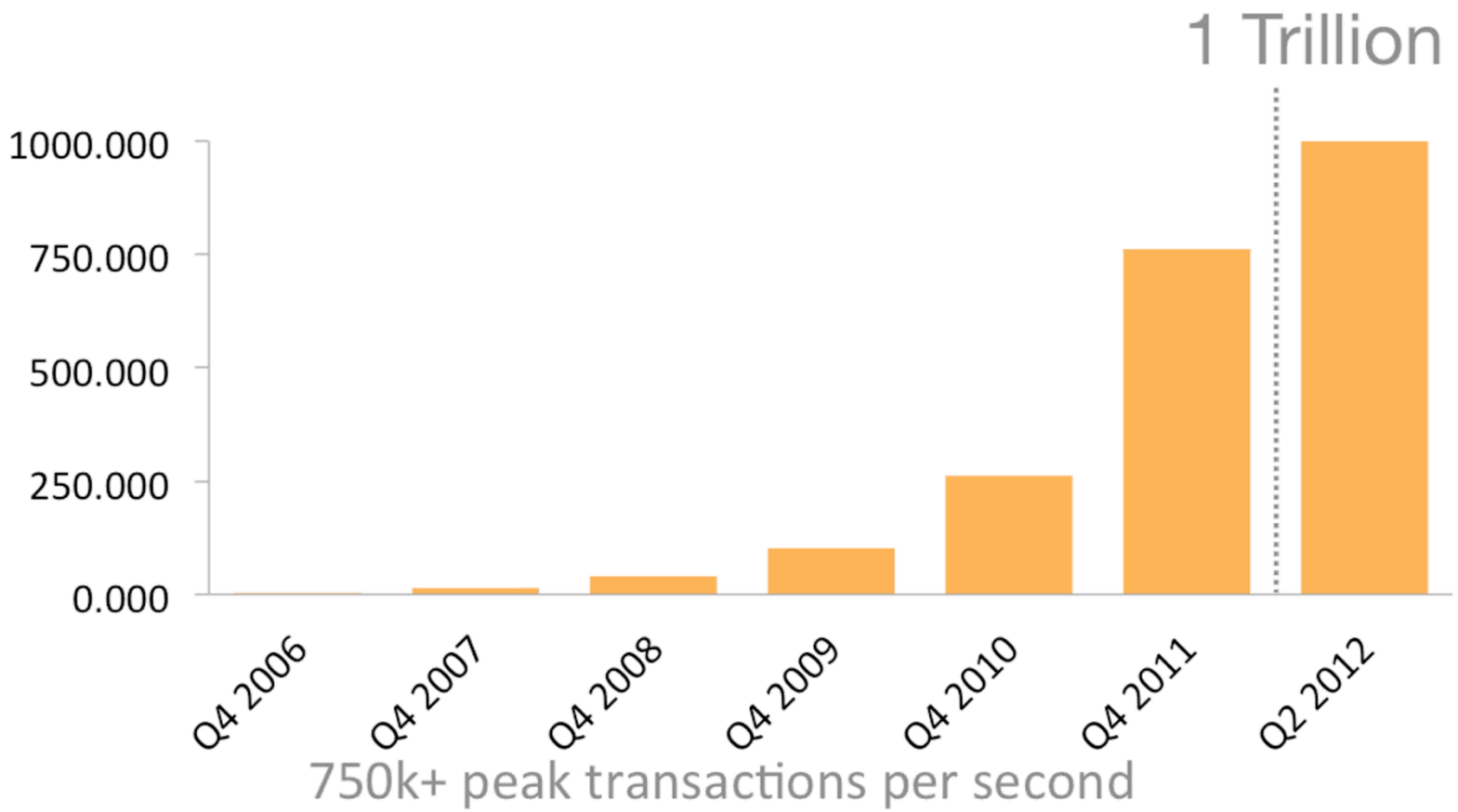
- **It is possible to get 12.7 Teraflops for less than $35/hour !**

AWS: Simple Storage Service (S3)

- **Amazon S3 is storage for the Internet. It is designed to make web-scale computing easier for developers. Amazon S3 provides a simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web. It gives any developer access to the same highly scalable, reliable, secure, fast, inexpensive infrastructure that Amazon uses to run its own global network of web sites. The service aims to maximize benefits of scale and to pass those benefits on to developers.**

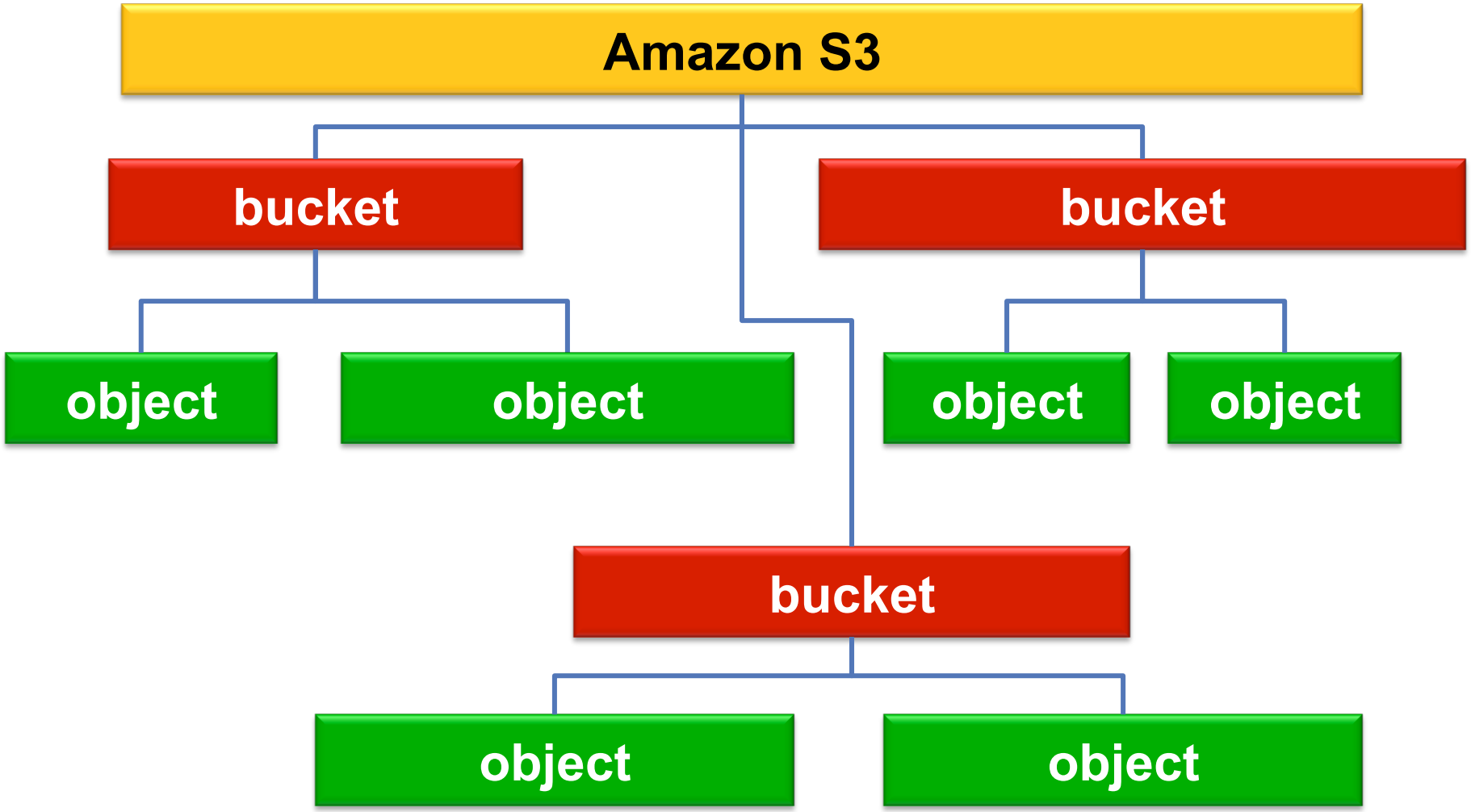
- **Features**
  - Distributed, replicated object store
  - Objects are stored in “Buckets”
    - Buckets are stored in an AWS region and replicated across AZs
    - Store anything…pictures, XML docs, encrypted blobs
    - 99.999999999% durability
    - Actual status: ~1 trillion objects and > 700,000 requests/second
  - AWS Import/Export Service
  - AWS Storage Gateway

Objects in S3

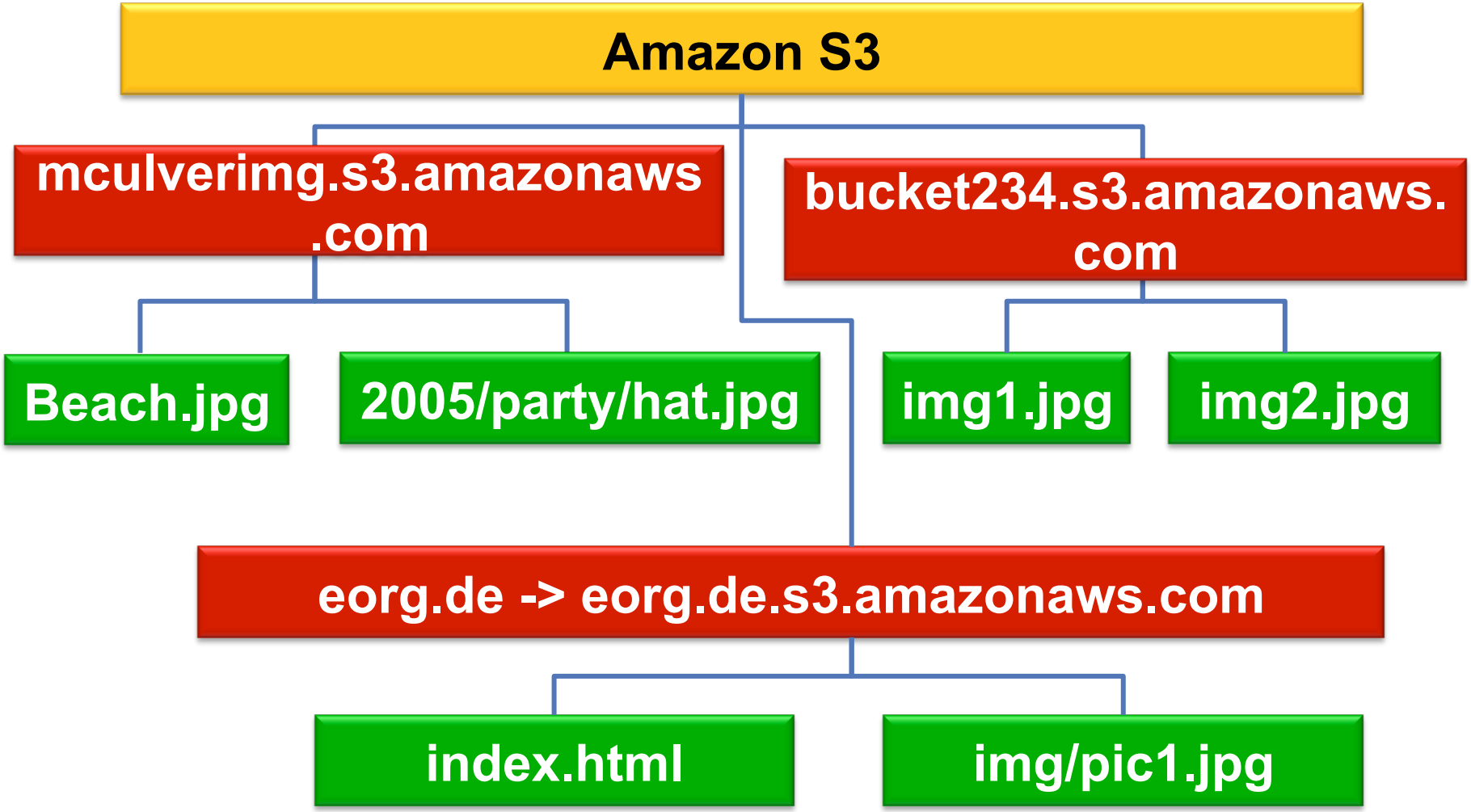


Source: aws.amazon.com

AWS: S3 Namespace



AWS: S3 Namespace



AWS: S3 Pricing

Storage Pricing

Region: US Standard

	Standard Storage	Reduced Redundancy Storage
First 1 TB / month	$0.125 per GB	$0.093 per GB
Next 49 TB / month	$0.110 per GB	$0.083 per GB
Next 450 TB / month	$0.095 per GB	$0.073 per GB
Next 500 TB / month	$0.090 per GB	$0.063 per GB
Next 4000 TB / month	$0.080 per GB	$0.053 per GB
Over 5000 TB / month	$0.055 per GB	$0.037 per GB

Request Pricing

Region: US Standard

	Pricing
PUT, COPY, POST, or LIST Requests	$0.01 per 1,000 requests
GET and all other Requests †	$0.01 per 10,000 requests
† No charge for delete requests	

Source: aws.amazon.com, 2012

AWS: S3 Pricing

Data Transfer Pricing

Region: US Standard

	Pricing
Data Transfer IN	
All data transfer in	$0.000 per GB
Data Transfer OUT	
First 1 GB / month	$0.000 per GB
Up to 10 TB / month	$0.120 per GB
Next 40 TB / month	$0.090 per GB
Next 100 TB / month	$0.070 per GB
Next 350 TB / month	$0.050 per GB
Next 524 TB / month	Contact Us
Next 4 PB / month	Contact Us
Greater than 5 PB / month	Contact Us

Source: aws.amazon.com, 2012

AWS: Elastic Block Store (EBS)

- **Amazon Elastic Block Store (EBS) provides block level storage volumes for use with Amazon EC2 instances. Amazon EBS volumes are network-attached, and persist independently from the life of an instance. Amazon EBS provides highly available, highly reliable, predictable storage volumes that can be attached to a running Amazon EC2 instance and exposed as a device within the instance. Amazon EBS is particularly suited for applications that require a database, file system, or access to raw block level storage.**

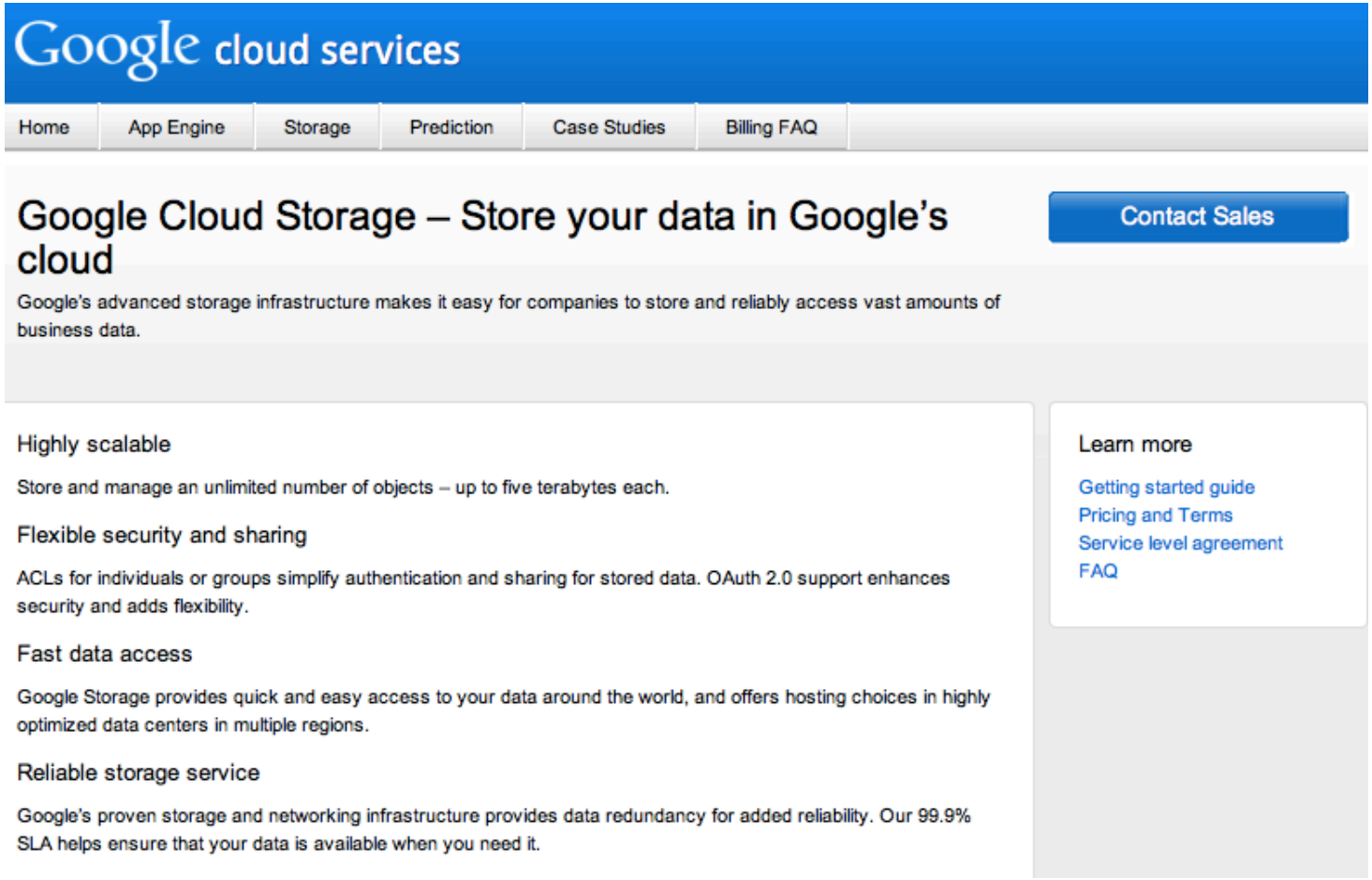
- **Features**
  - Persistent data storage
  - Works like an unformatted storage device (like an USB disk drive)
  - Can be dynamically mapped to EC2 instance
  - Can take snapshots to conserve a specific status (backup)
  - Pricing is the same as for S3

AWS Glacier: Archive as a Service

- **Amazon Glacier is an extremely low-cost storage service that provides secure and durable storage for data archiving and backup. In order to keep costs low, Amazon Glacier is optimized for data that is infrequently accessed and for which retrieval times of several hours are suitable. With Amazon Glacier, customers can reliably store large or small amounts of data for as little as $0.01 per gigabyte per month, a significant savings compared to on-premises solutions.**

- **Common Use Cases**
  - Offsite enterprise information archiving
  - Archiving media assets
  - Archiving research and scientific data
  - Digital preservation
  - Magnetic tape replacement

Google Cloud Storage



- **API is compatible to S3 (S3 may be regarded a de-facto standard)**
- **http://www.google.com/enterprise/cloud/storage/**

Google Cloud Storage

Monthly Usage	Price (per GB)
First 0 - 1TB	$0.12
Next 9TB	$0.105
Next 90TB	$0.095
Next 400TB	$0.085
Additional Storage	Contact us

Network

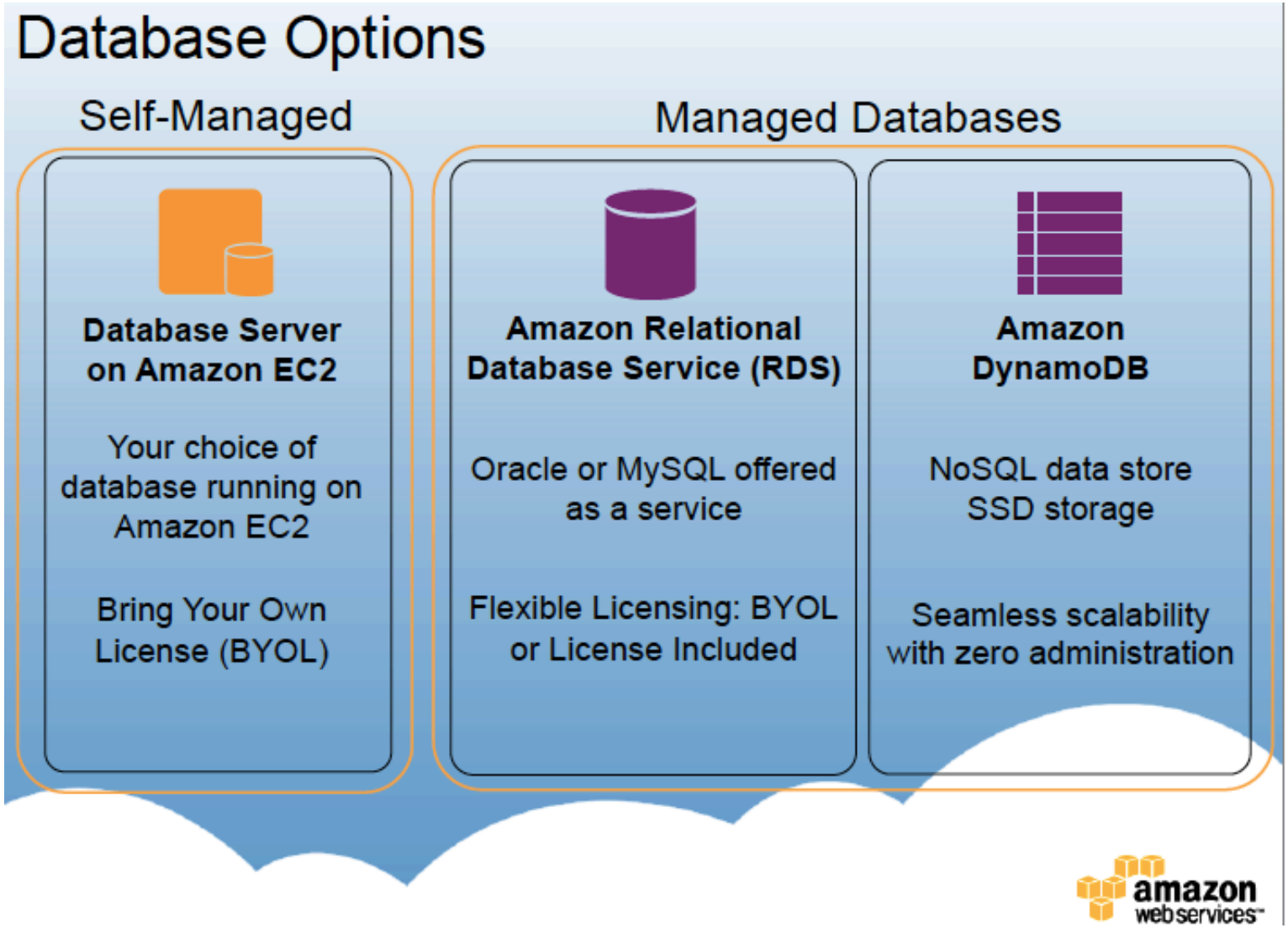
Monthly Usage	Network (Egress) - Americas and EMEA* (per GB)	Network (Egress) - Asia-Pacific (per GB)	Network (Ingress)
0 - 1TB	$0.12	$0.21	Free
Next 9TB	$0.11	$0.18	
Next 90TB	$0.08	$0.15	
Additional Data Transfer	Contact us		

Requests

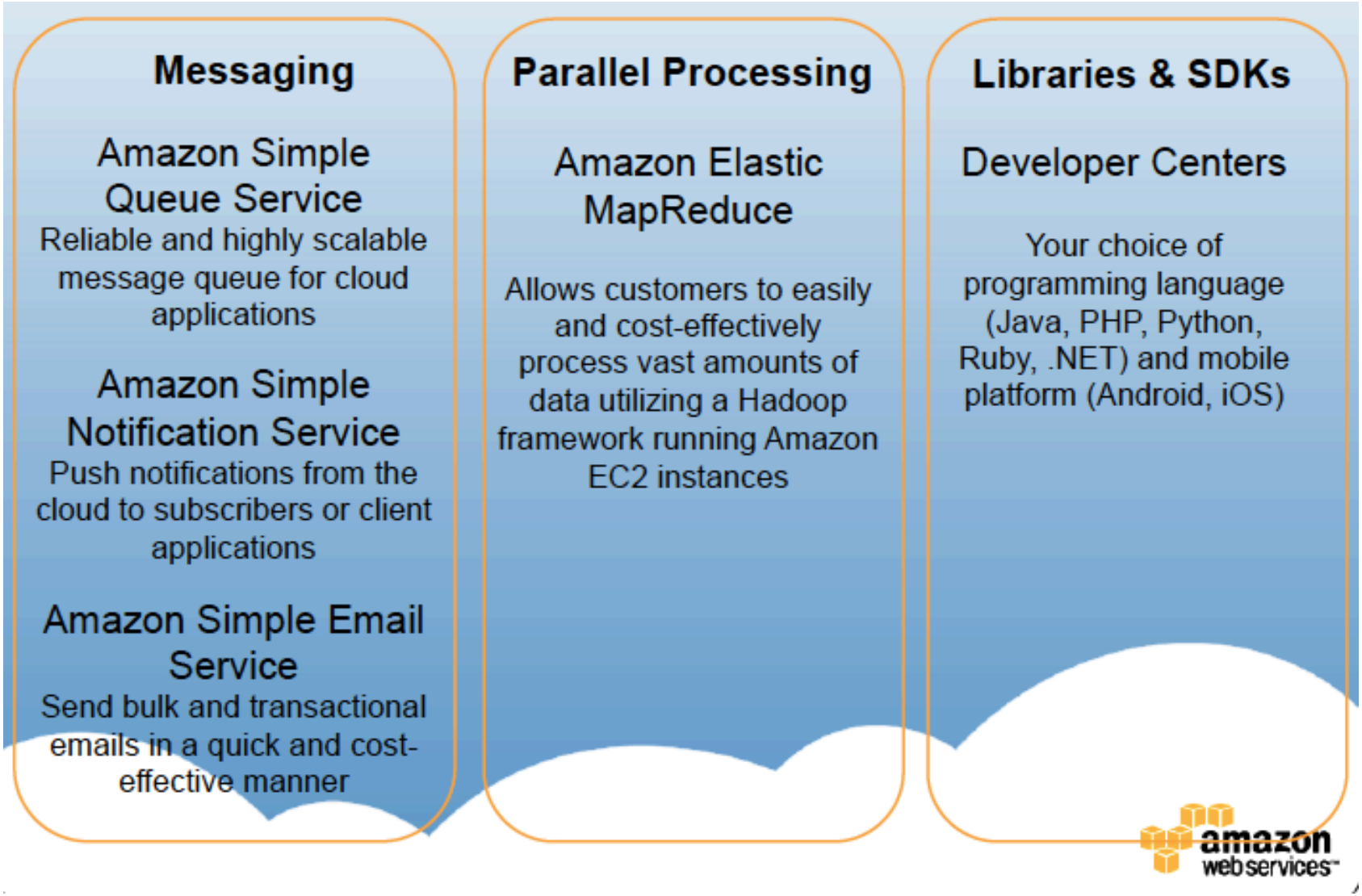
PUT, POST, GET bucket**, GET service** Requests (per 1,000 requests/month)	GET, HEAD Requests (per 10,000 requests/month)
$0.01	$0.01

- **Pricing is almost compatible to S3 (a little bit cheaper than S3)**

AWS Database Services

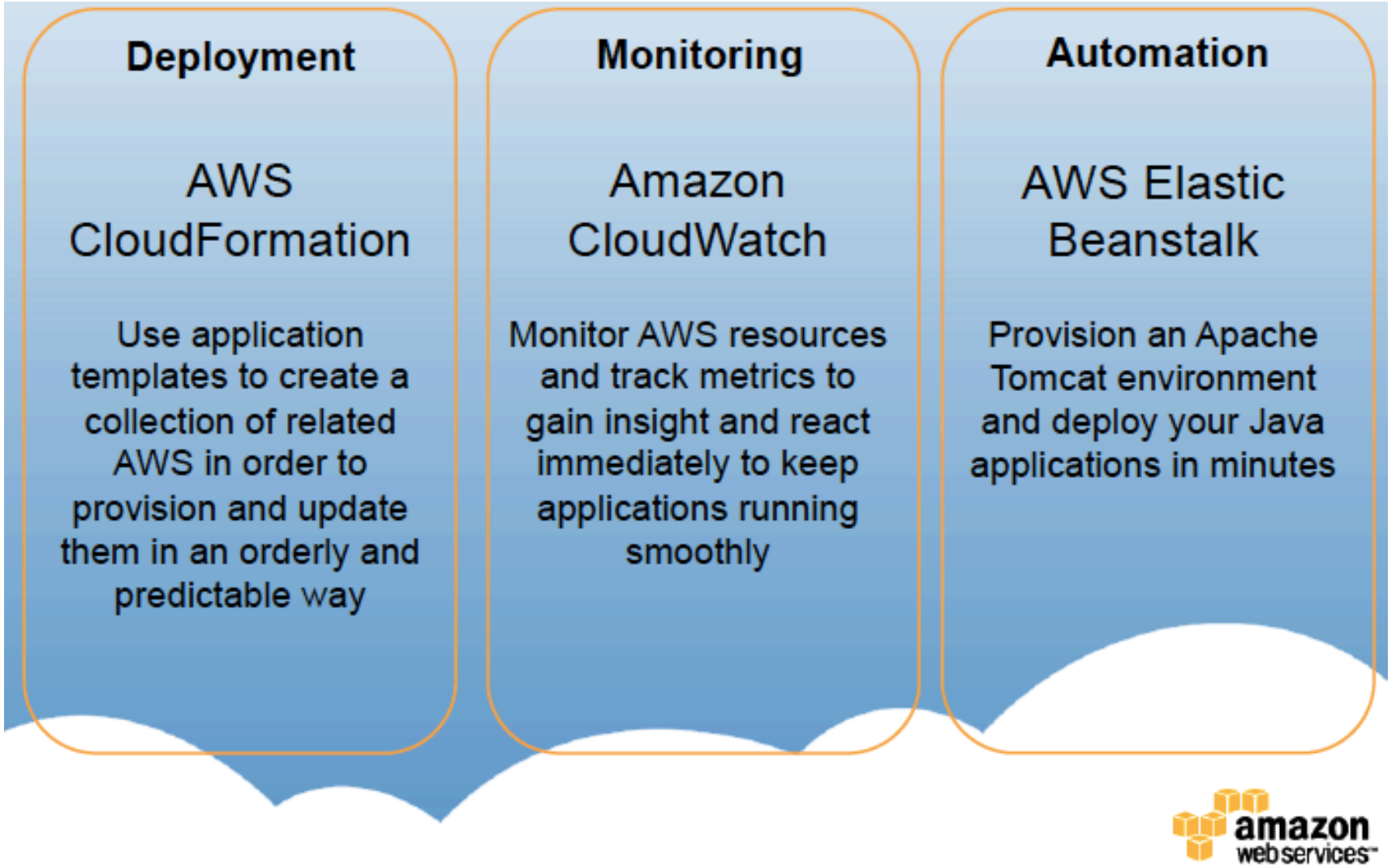


AWS Higher Level Services



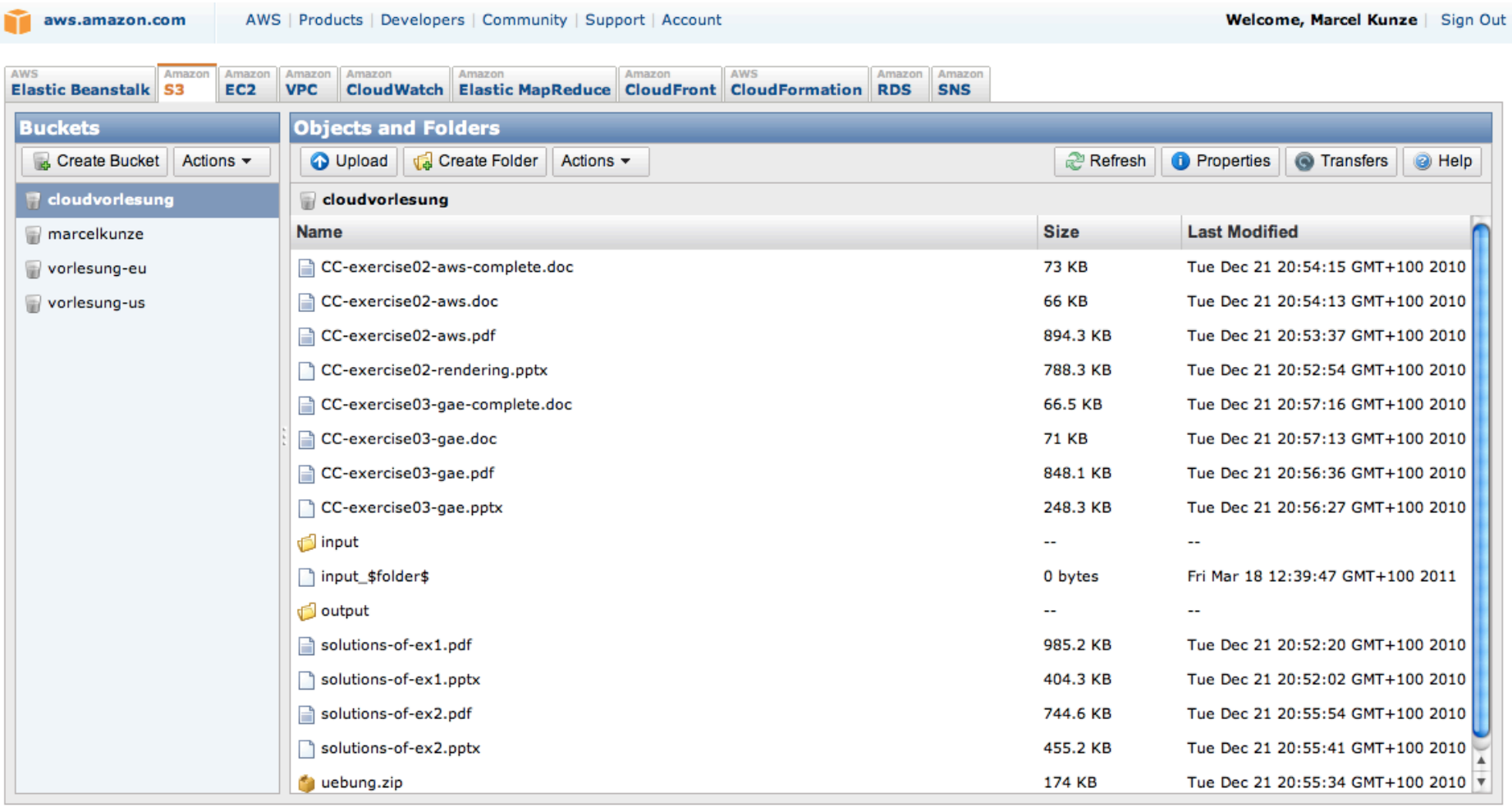
AWS Deployment & Administration Services

Karlsruhe Institute of Technology



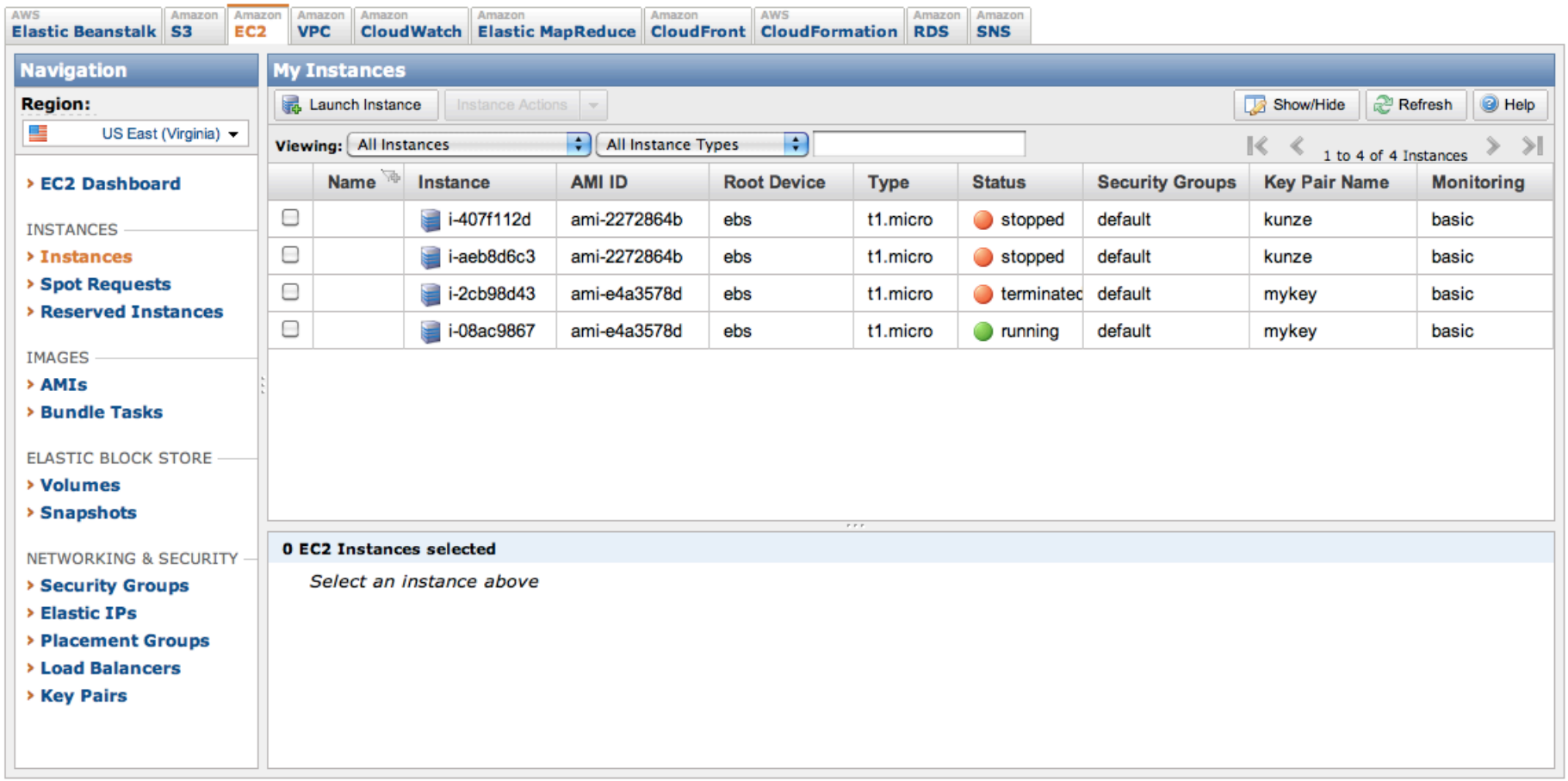
3. Cloud Management

AWS Management Console (1)



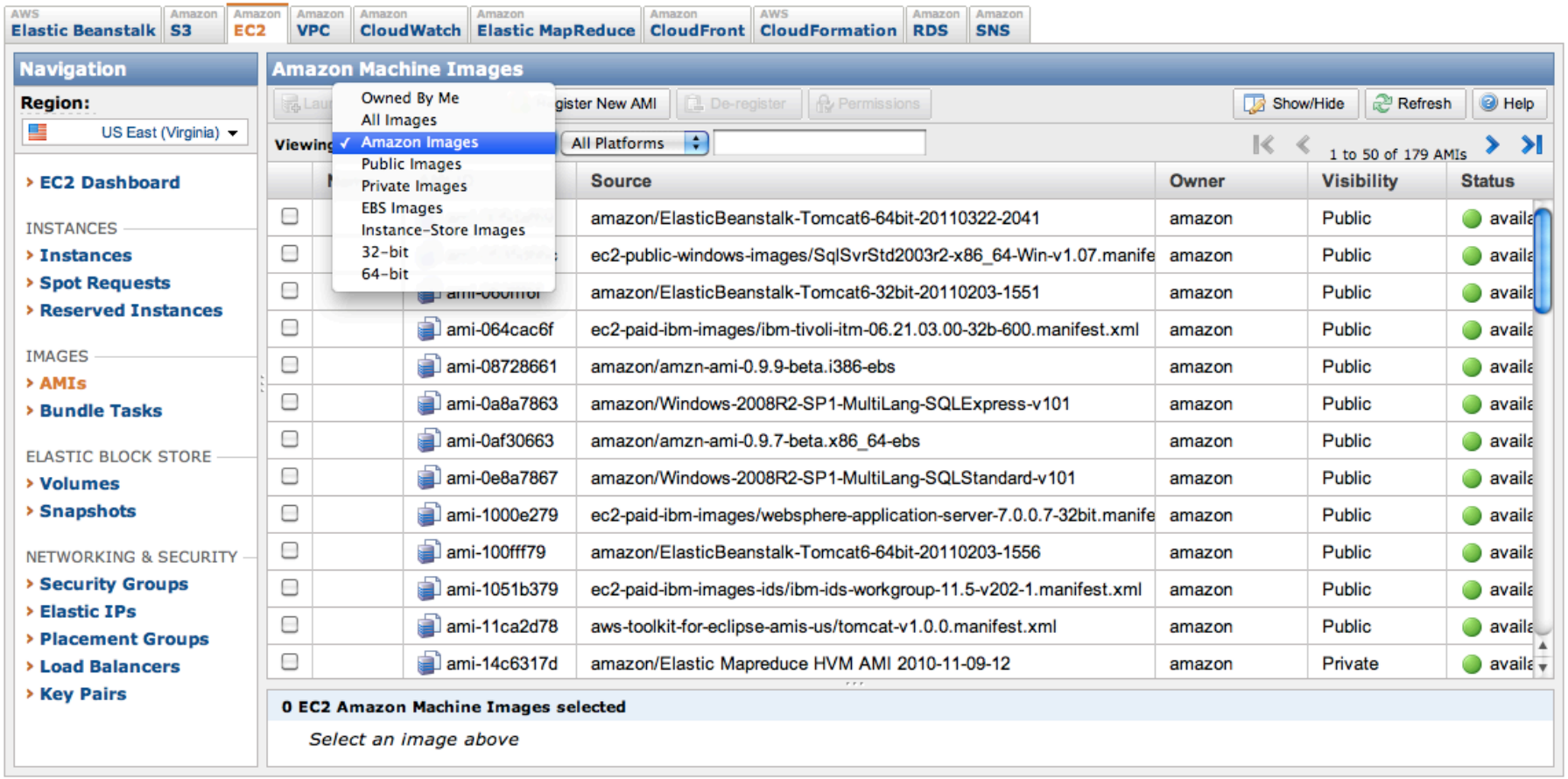
- **Management of virtual infrastructure**
- https://console.aws.amazon.com/

AWS Management Console (2)



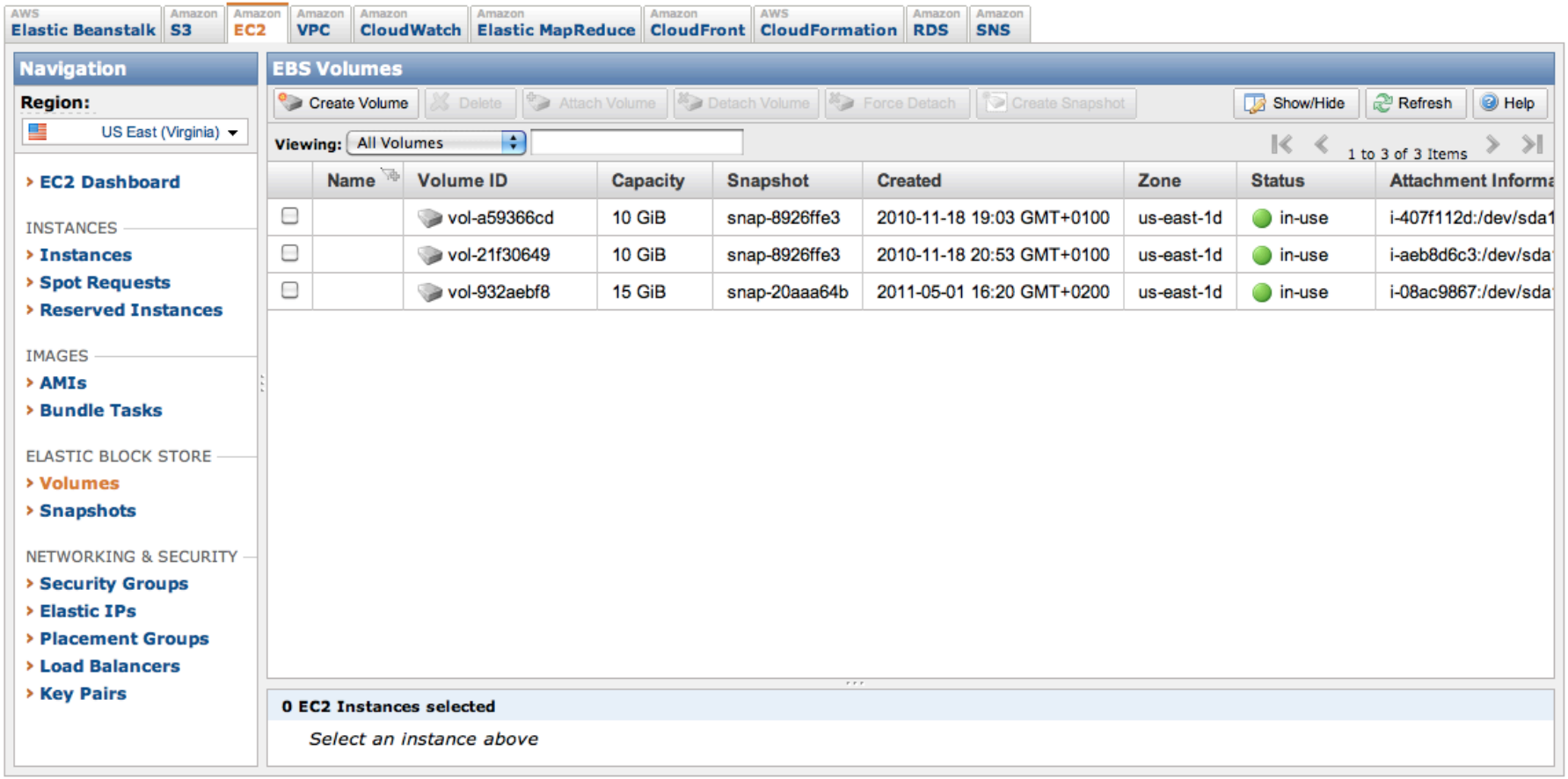
- **Management of EC2 instances**

AWS Management Console (3)



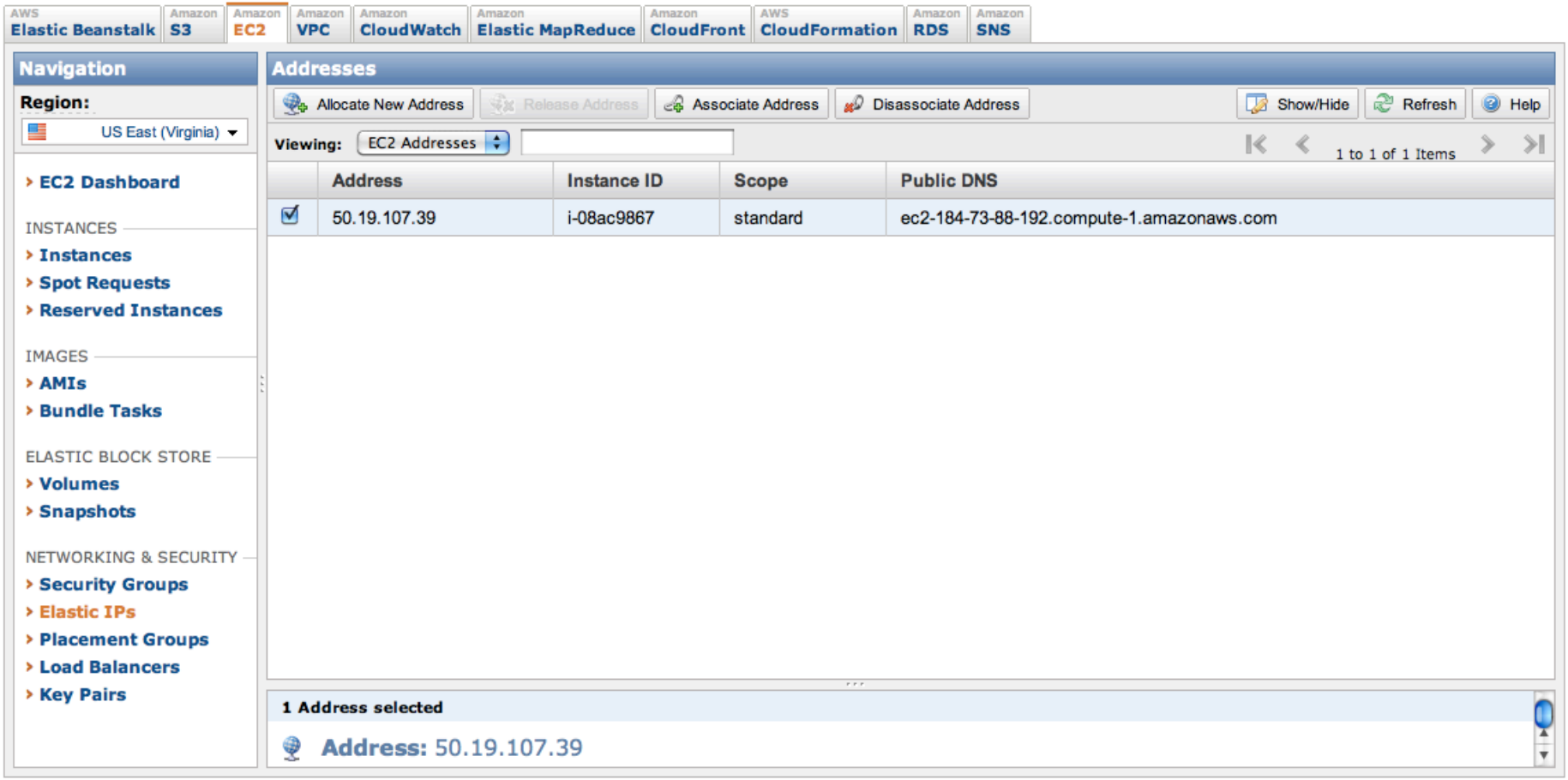
- **Management of Amazon Machine Images (AMI)**

AWS Management Console (4)



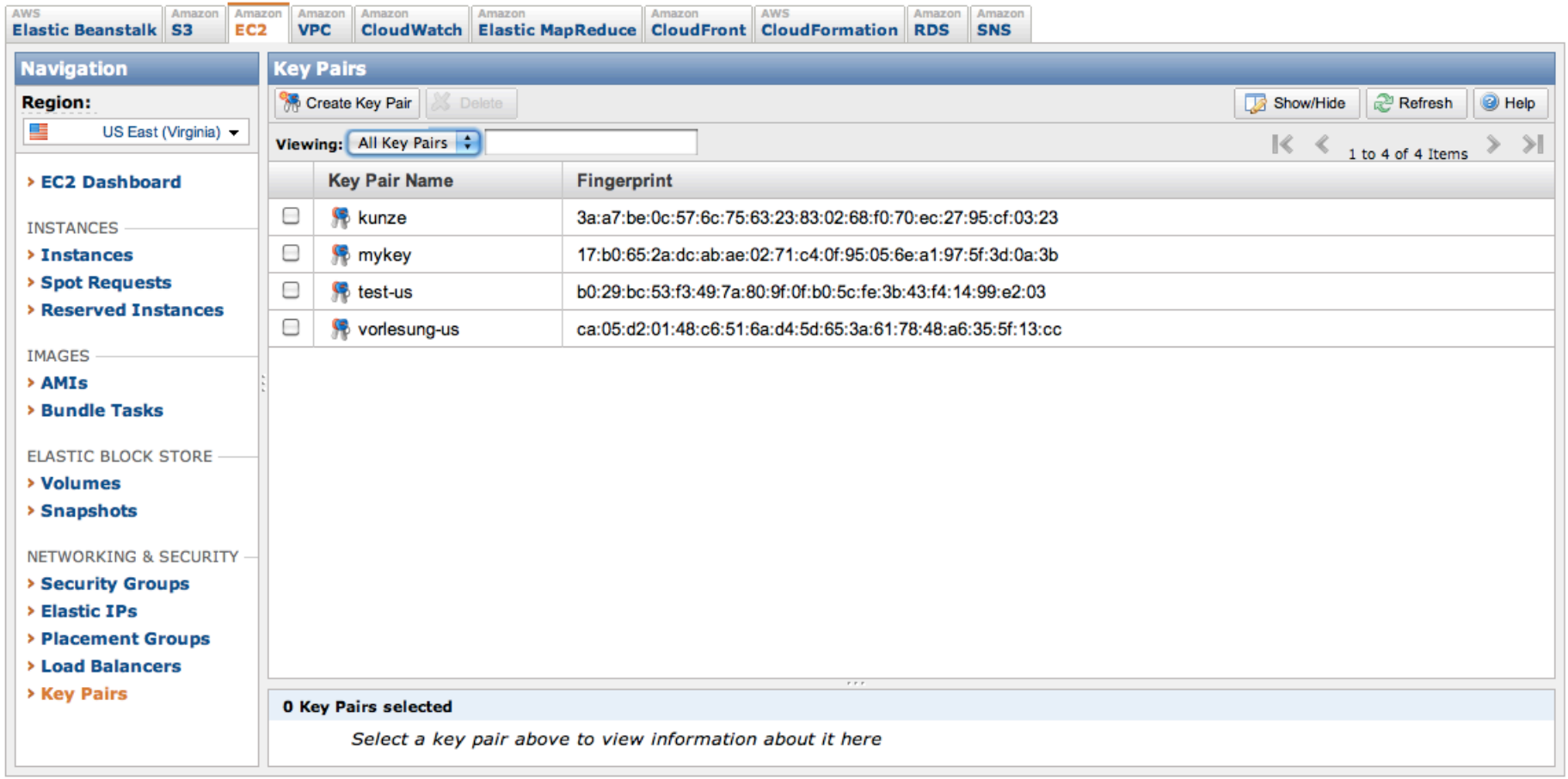
- **Storage management (Elastic Block Store)**

AWS Management Console (5)



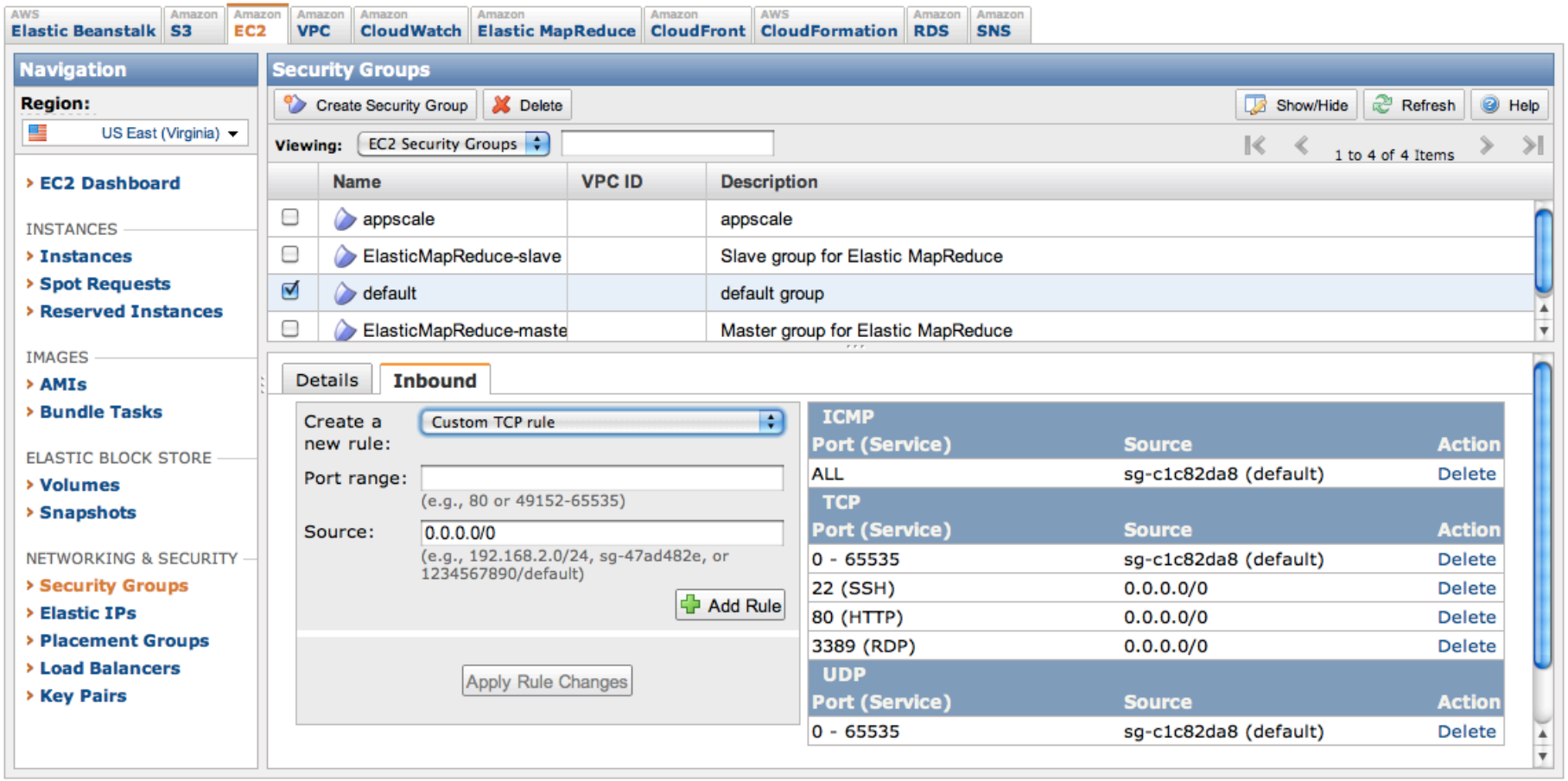
- Management of public IP adresses: Elastic IP
- Dynamic allocation to EC2 instances

AWS Management Console (6)



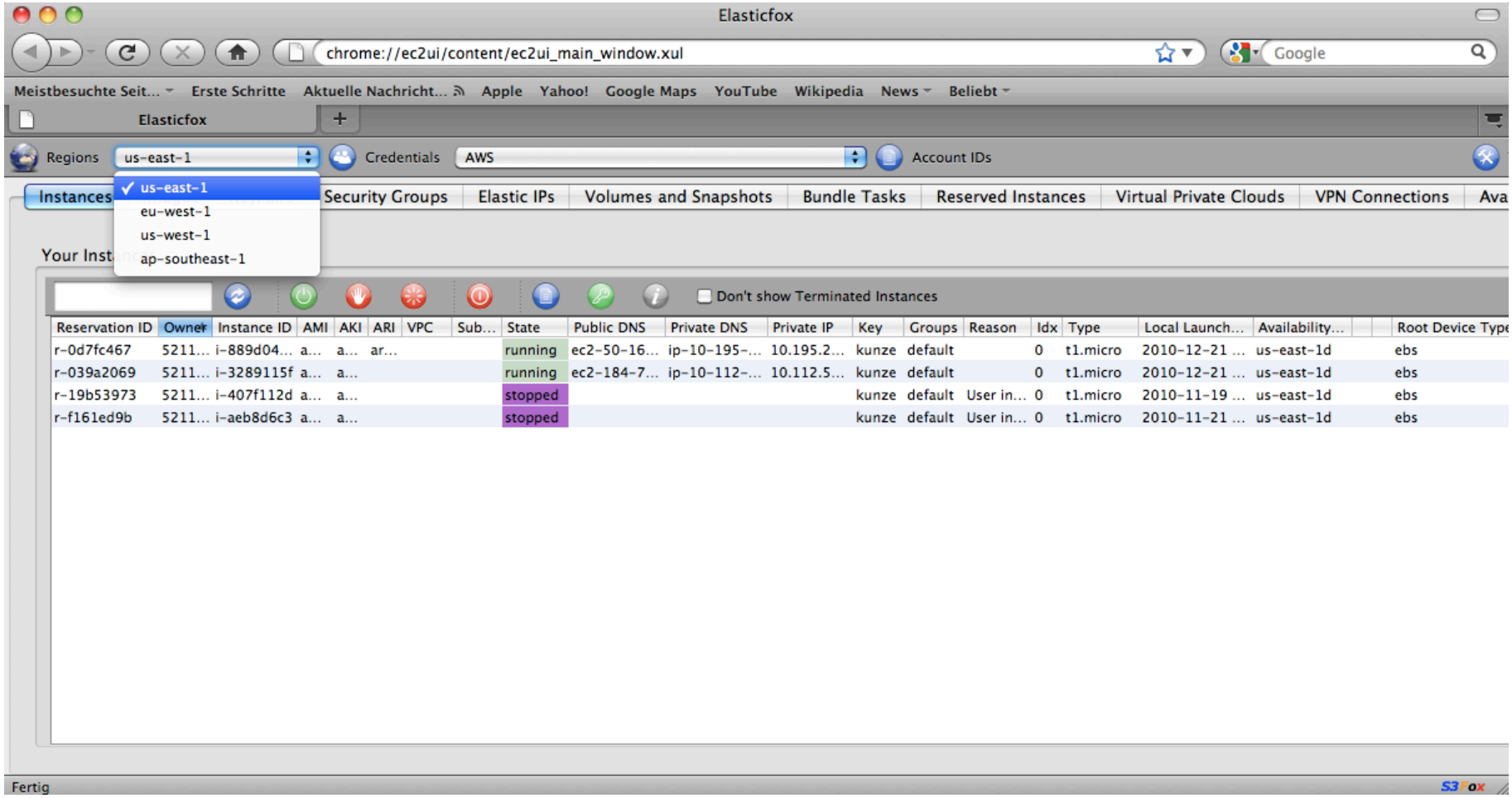
- **Management of key pairs (Privileged access via SSH/RDP)**

AWS Management Console (7)



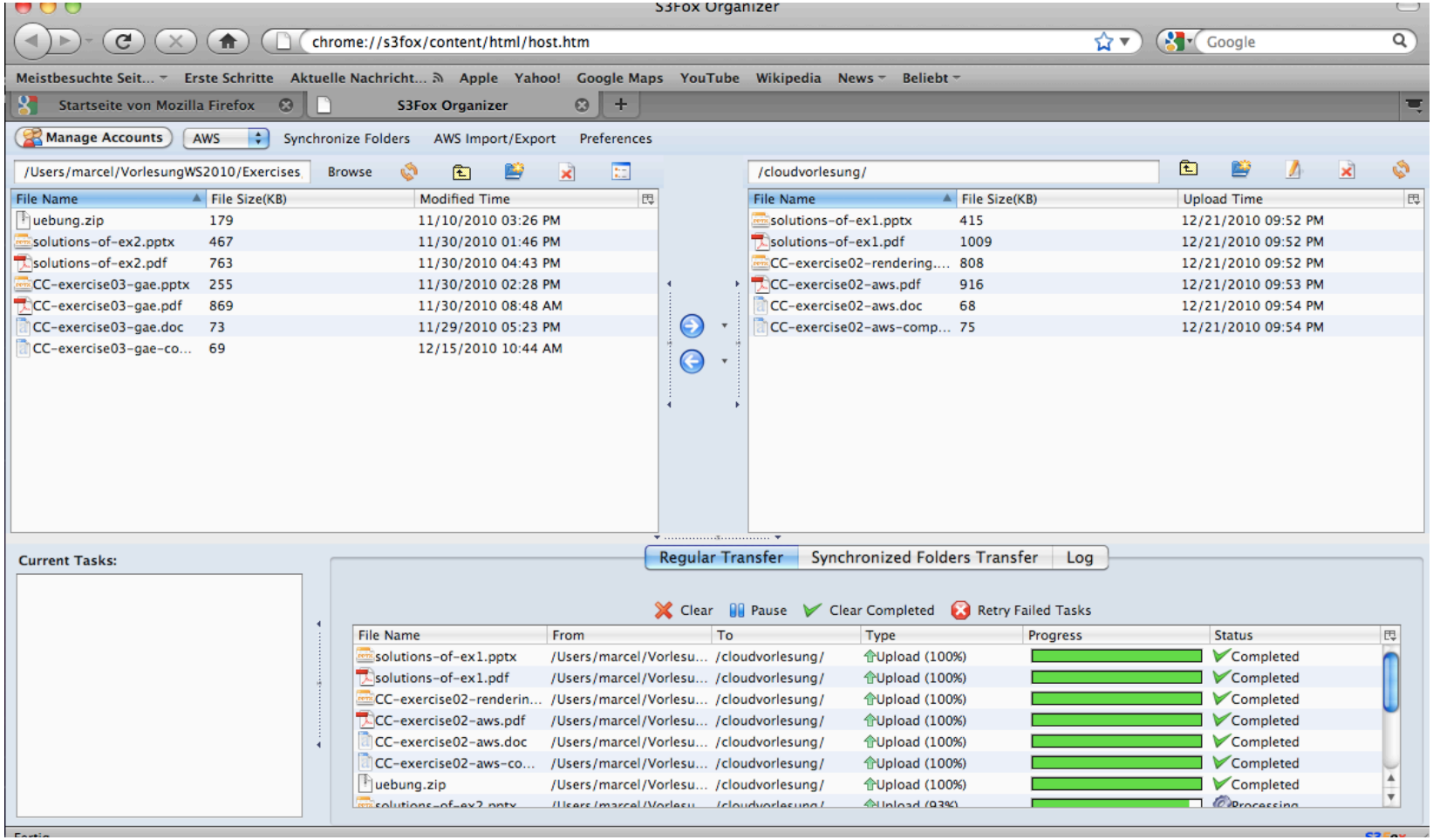
- **Security Groups: Administration of firewall rules**

Browser Plugin: ElasticFox



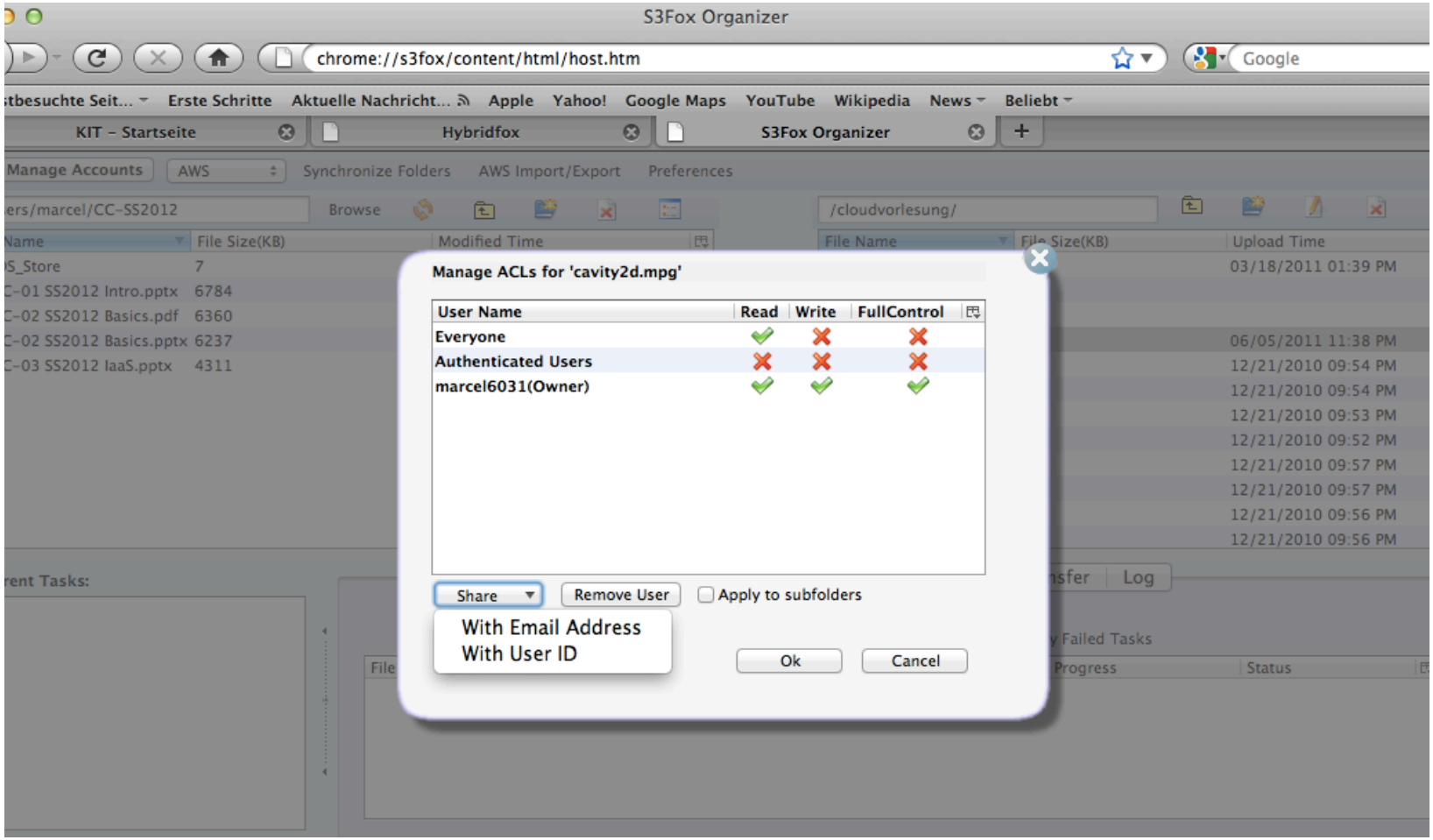
- **ElasticFox FireFox Plugin**
- **Download at http://aws.amazon.com/developertools/609**

Browser Plugin: S3Fox



- **S3Fox FireFox Plugin**
- **Download at** http://www.s3fox.net/

Access Control Lists (ACL)



- Define access rights: Read, write, fullcontrol
- Add specific users with e-mail or user ID

Available on the iPhone App Store

iAWSManager

Manage your Amazon Web services from your iPhone

Available on the iPhone App Store

Karlsruhe Institute of Technology

Home | Screenshots | Feedback | S3Cloud | Disclaimer

Welcome to iAWSManager.

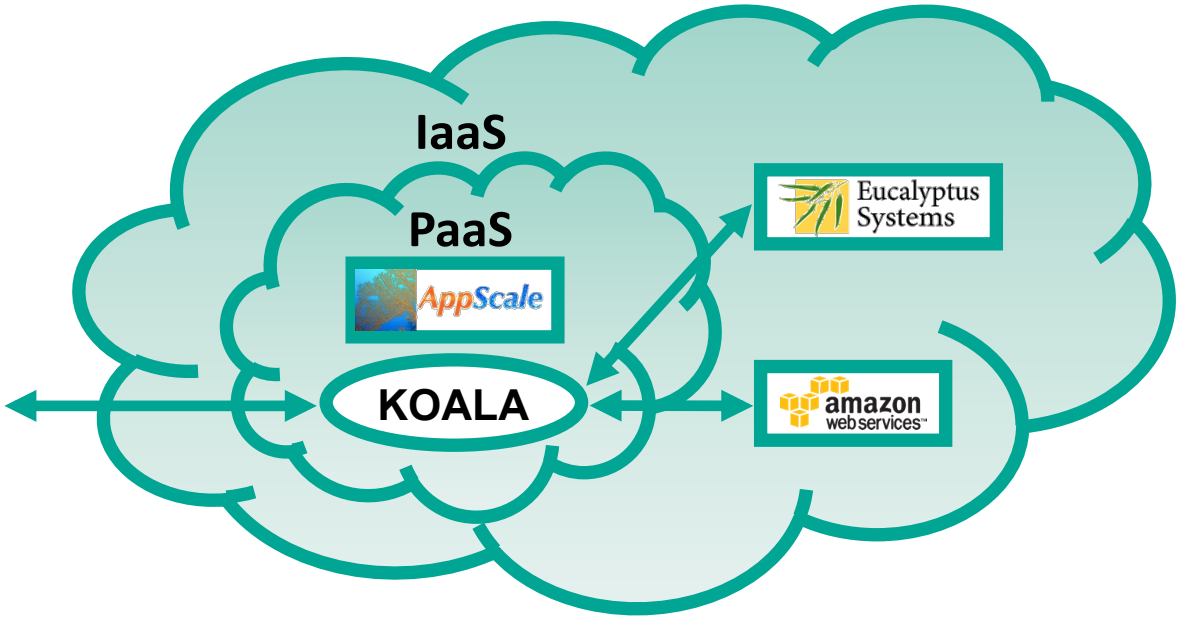
iAWSManager is an iPhone Application to manage your Amazon Web Services.You can manage Your Amazon EC2 S3 CloudFront SQS SDB resources from your iPhone and iPod Touch.

Main Features

- **EC2:**
  - Manage instance operations (Launch,Terminate,Reboot,Console,Assign elastic IP, Attach/Detach Volume)
  - Manage ElasticIP,Security Groups,Key pairs
  - Create Volumes
  - Create snapshots from volumes
  - Manage your EC2 instances in all Regions
  - Search Available EC2 images
  - Color coded server status messages
  - Start/Stop Servers
  - Cloud watch Monitoring
  - Elastic load balancers support

KOALA Cloud Manager

http://koalacloud.appspot.com

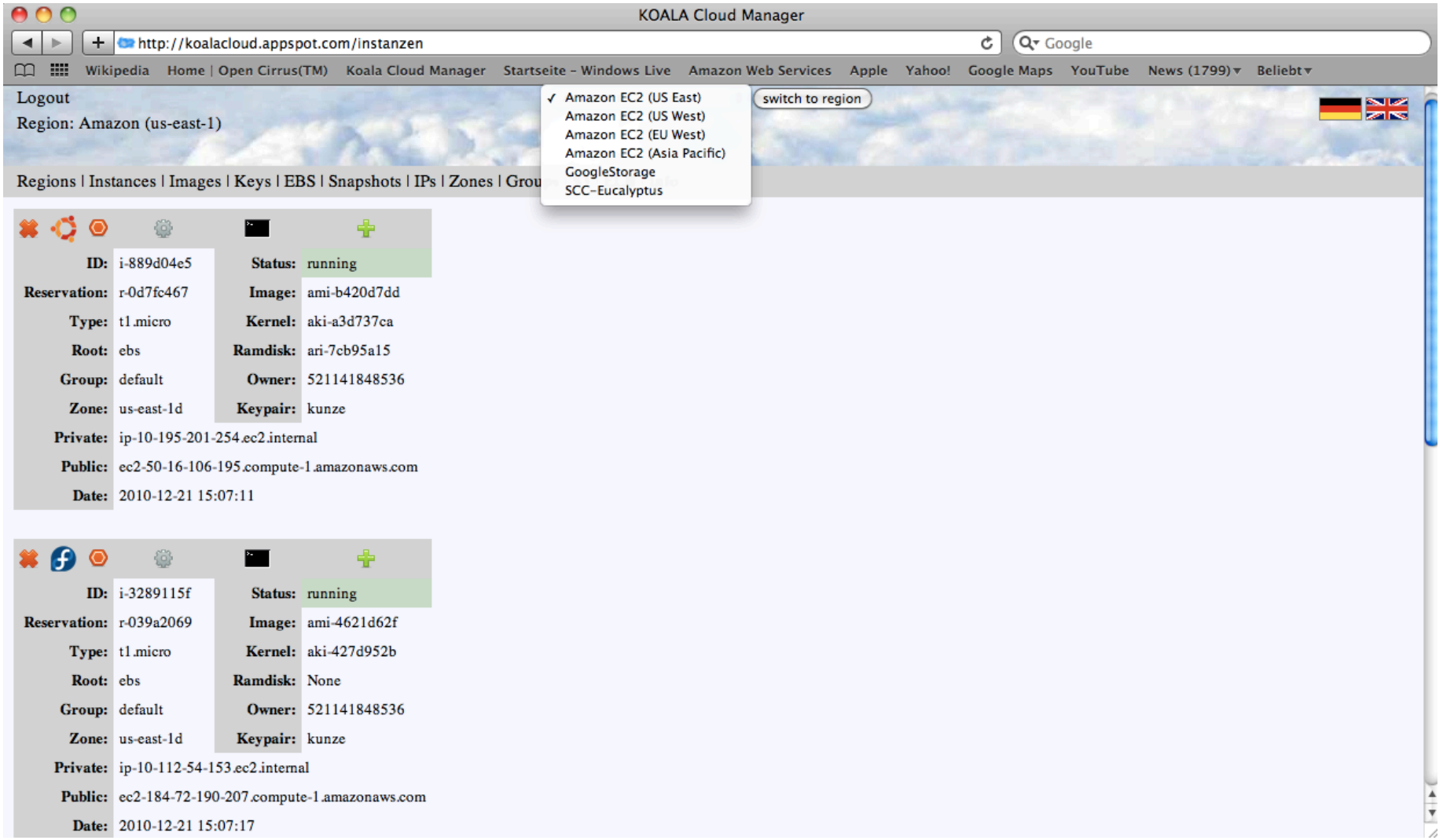


"Cloud Architecture"

- **Mobile management of hybrid cloud resources as SaaS solution running on PaaS on top of an IaaS (In any combination with various providers!)**
- Christian Baun, Marcel Kunze, Viktor Mauch: The KOALA Cloud Manager: Cloud Service Management the Easy Way. IEEE CLOUD 2011: 744-745

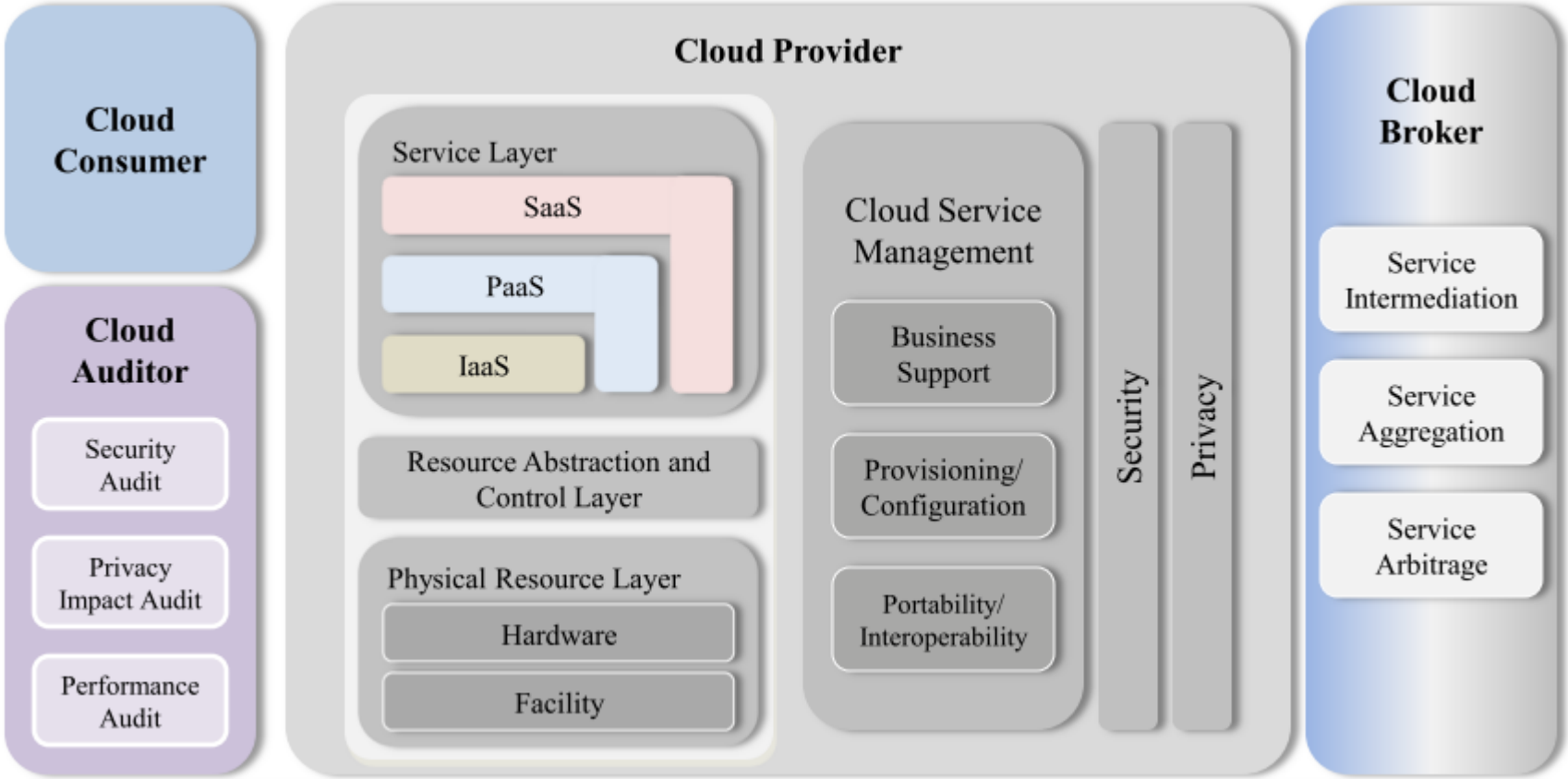
Cloud Service: KOALA

http://koalacloud.appspot.com - http://code.google.com/p/koalacloud



4. Cloud Architecture

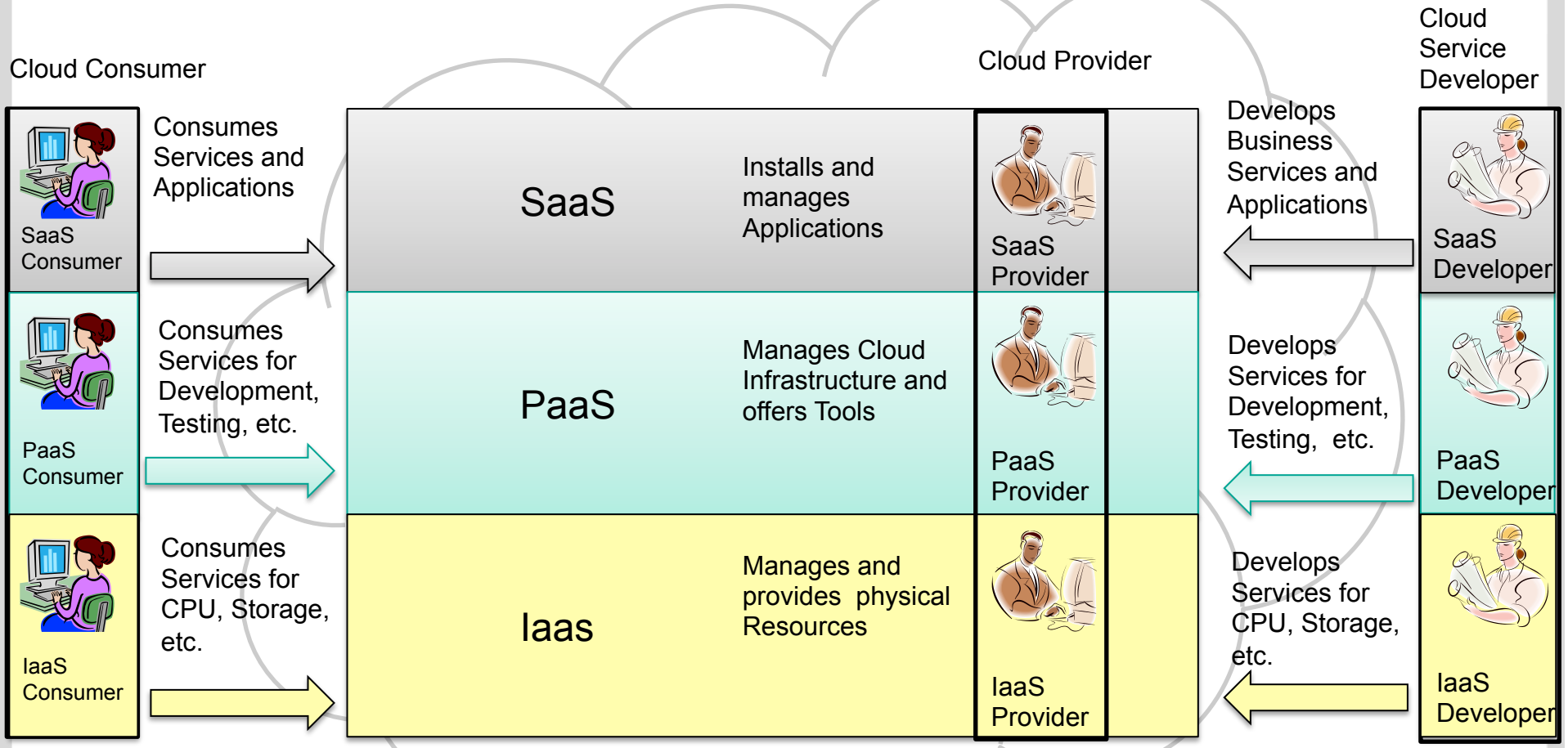
NIST Reference Architecture



- **NIST Cloud Computing Reference Architecture, V1, March 30, 2011, p.4**

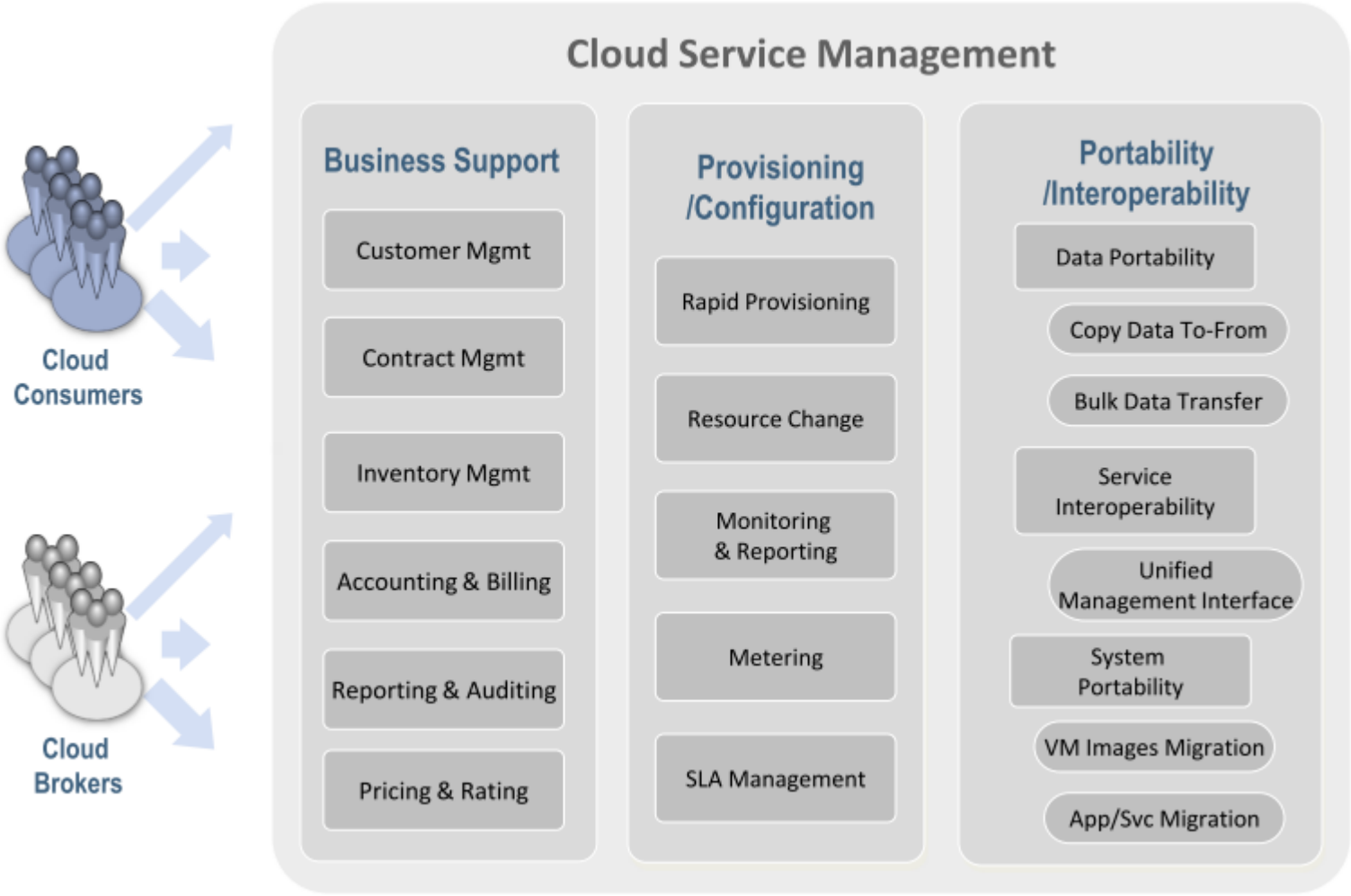
http://collaborate.nist.gov/twiki-cloud-computing/pub/CloudComputing/ReferenceArchitectureTaxonomy/NIST_CC_Reference_Architecture_v1_March_30_2011.pdf

NIST Reference Architecture: Roles



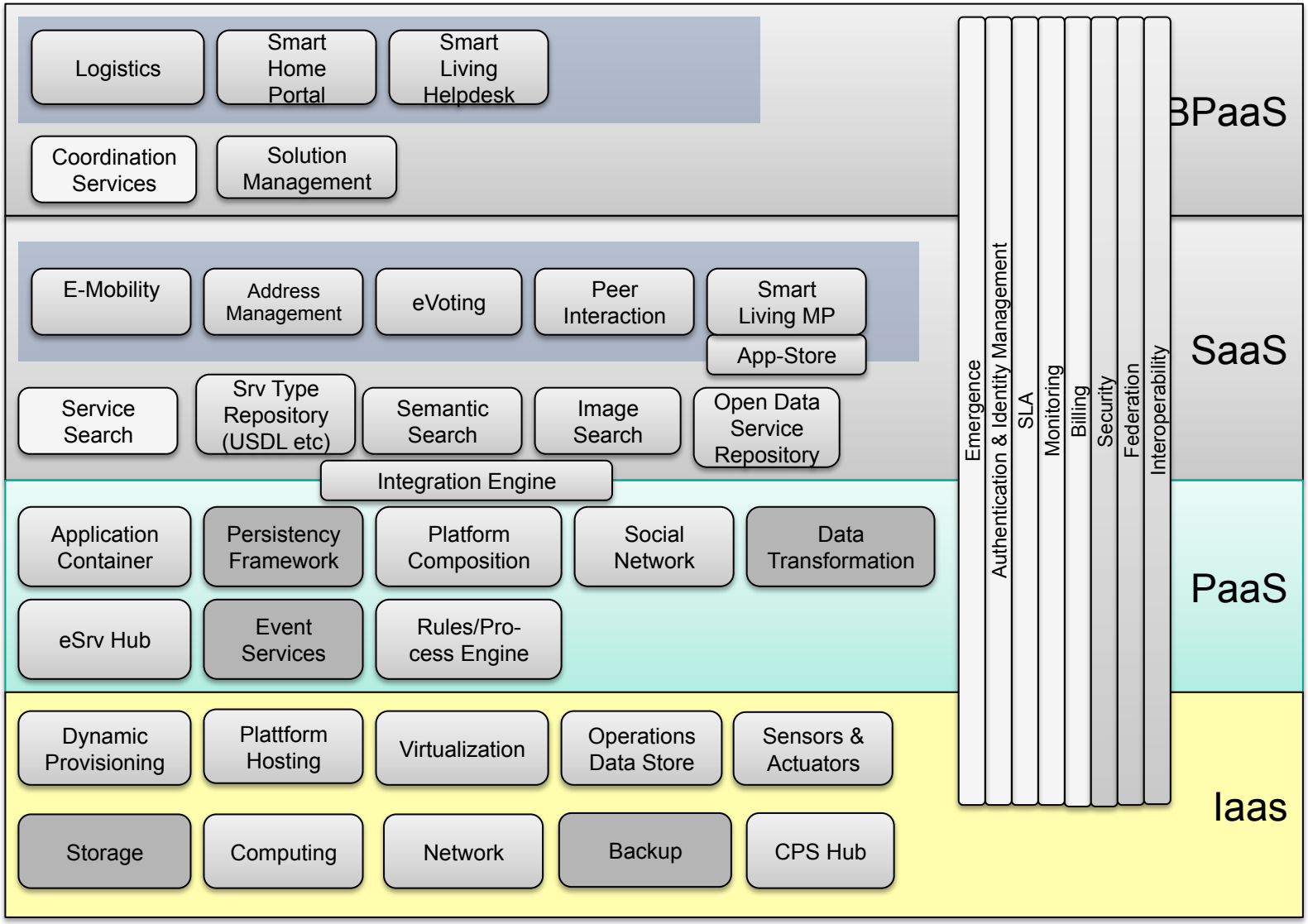
Source: SAP

NIST Cloud Service Management



Example: Software Cluster Architecture

http://www.software-cluster.com



Source: SAP

5. Programming Models

Management

- **AWS command line tools are useful for developers and admins**
  - Difficult to use for average customer
  - Download from http://aws.amazon.com/developertools/351

Amazon Elastic Compute Cloud (Version 2011-02-28)

Revised: 3/25/2011

Quick Reference Card (Page 1)

Karlsruhe Institute of Technology

Instance Tools

ec2-run-instances (ec2run)
Launches one or more instances of the specified AMI.
`ec2-run-instances ami_id [-n instance_count] [-g group [-g group ...]] [-k keypair] [-d user_data | -f user_data_file] [-t instance_type] [-z availability_zone] [--kernel kernel_id] [--ramdisk ramdisk_id] [-b block-device-mapping] [--monitor] [--disable-api-termination] [--instance-initiated-shutdown-behavior behavior] [--placement-group placement_group] [--tenancy tenancy] [-s subnet] [--private-ip-address ip_address]`

ec2-describe-instances (ec2din)
Lists the specified instances. If no instance is specified, all your instances are listed.
`ec2-describe-instances [instance_id ...] [[--filter name=value] ...]`

ec2-terminate-instances (ec2kill)
Terminates the specified instance.
`ec2-terminate-instances instance_id [instance_id ...]`

ec2-start-instances (ec2start)
Starts a stopped instance (applies only to Amazon EBS-backed instances).
`ec2-start-instances instance_id [instance_id ...]`

ec2-describe-instance-attribute (ec2dinatt)
Describes an attribute for the specified instance.
`ec2-describe-instance-attribute instance_id { --block-device-mapping | --disable-api-termination | --instance-initiated-shutdown-behavior | --instance-type | --kernel | --ramdisk | --root-device-name | --user-data }`

ec2-stop-instances (ec2stop)
Stops an instance (applies only to Amazon EBS-backed instances).
`ec2-stop-instances instance_id [instance_id ...]`

ec2-monitor-instances (ec2min)
Enables monitoring for the specified instance.
`ec2-monitor-instances instance-id [instance_id ...]`

ec2-unmonitor-instances (ec2umin)
Disables monitoring for the specified instance(s).
`ec2-monitor-instances instance-id [instance_id ...]`

Amazon EBS Tools

ec2-create-volume (ec2addvol)
Creates an Amazon EBS volume from a snapshot or an empty volume in the size you specify.
`ec2-create-volume -z availability_zone [-s size | --snapshot snapshot]`

ec2-describe-volumes (ec2dvol)
Lists the specified Amazon EBS volumes. If no volume is specified, all your volumes are listed.
`ec2-describe-volumes [volume_id ...] [[--filter name=value] ...]`

ec2-delete-volume (ec2delvol)
Deletes the specified Amazon EBS volume.
`ec2-delete-volume volume_id`

ec2-attach-volume (ec2attvol)
Attaches an Amazon EBS volume to a running instance and exposes it as the specified device. The volume and instance must be in the same Availability Zone.
`ec2-attach-volume volume_id -i instance_id -d device`

ec2-create-snapshot (ec2addsnap)
Creates a snapshot of an Amazon EBS volume and stores it in Amazon S3.
`ec2-create-snapshot volume_id`

ec2-describe-snapshots (ec2dsnap)
Lists the specified snapshots. If no snapshot is specified, all your snapshots are listed.
`ec2-describe-snapshots [snapshot_id ...] [[--filter name=value] ...]`

ec2-delete-snapshot (ec2delsnap)
Deletes the specified Amazon EBS snapshot.
`ec2-delete-snapshot snapshot_id`

ec2-detach-volume (ec2detvol)
Detaches an Amazon EBS volume from an instance.
`ec2-detach-volume volume_id [-i instance_id [-d device]] [--force]`

EC2 Elastic IP Address Tools

ec2-allocate-address (ec2allocaddr)
Acquires an EC2 Elastic IP address for use with your account.
`ec2-allocate-address`

ec2-describe-addresses (ec2daddr)
Lists both EC2 and VPC Elastic IP addresses assigned to your account.
`ec2-describe-addresses [ip_address ...]`

ec2-release-address (ec2reladdr)
Releases an EC2 Elastic IP address associated with your account.
`ec2-release-address ip_address`

ec2-associate-address (ec2assocaddr)
Associates an EC2 Elastic IP address with an instance. If the IP address is currently assigned to another instance, the IP address is reassigned to the specified instance.
`ec2-associate-address ip_address -i instance_id`

ec2-disassociate-address (ec2disaddr)
Disassociates the specified EC2 Elastic IP address from the instance to which it is assigned.
`ec2-disassociate-address ip_address`

- http://awsdocs.s3.amazonaws.com/EC2/latest/ec2-qrc.pdf
- http://docs.amazonwebservices.com/AWSEC2/latest/GettingStartedGuide/

AWS Quick Reference Card (c'td)

Image Tools

ec2-describe-images (ec2dim)
Returns information about AMIs, AKIs, and ARIs. If no parameter is specified, information about all images for which you have launch permission is returned.
`ec2-describe-images [ami_id ...] [--all] [-o owner ...] [-x user_id] [[--filter name=value] ...]`

ec2-create-image (ec2cim)
Creates an AMI that uses an Amazon EBS root device from a running or stopped Amazon EBS-backed instance.
`ec2-create-image instance_id --name name [ --description description] [ --no-reboot]`

ec2-describe-image-attribute (ec2dimatt)
Describes an attribute for the specified AMI.
`ec2-describe-image-attribute ami_id { --launch-permission | --product-code | --block-device-mapping | --kernel | --ramdisk }`

ec2-register (ec2reg)
Registers the AMI specified in the manifest file and generates a new AMI ID.
`ec2-register manifest`

Availability Zone Tools

ec2-describe-availability-zones (ec2daz)
Lists Availability Zones that are currently available to your account.
`ec2-describe-availability-zones [zone ...]`

Windows Tools

ec2-get-password (ec2gpass)
Retrieves and decrypts the administrator password for the specified Windows instance.
`ec2-get-password instance_id -k key_pair`

ec2-bundle-instance (ec2bundle)

Key Pair Tools

ec2-create-keypair (ec2addkey)
Creates a new 2048-bit RSA key pair with the specified name.
`ec2-create-keypair key_pair`

ec2-describe-keypairs (ec2dkey)
Lists the specified key pairs. If no key pair is specified, all your key pairs are listed.
`ec2-describe-keypairs [key_pair ...]`

ec2-delete-keypair (ec2delkey)
Deletes the specified key pair by removing the public key from Amazon EC2.
`ec2-delete-keypair key_pair`

ec2-import-keypair (ec2ikey)
Imports the public key for a key pair. You keep the private key. The key pair works in all EC2 Regions.
`ec2-import-keypair key_pair --public-key-file file`

EC2 Security Group Tools

ec2-create-group (ec2addgrp)
Creates a new EC2 security group. Group names must be unique per account.
`ec2-create-group group_name -d description`

ec2-delete-group (ec2delgrp)
Deletes the specified EC2 security group.
`ec2-delete-group ec2_group_name_or_id`

ec2-describe-group (ec2dgrp)
Lists your EC2 and VPC security groups. If no security group is specified, all your security groups are listed.
`ec2-describe-group [ec2_group_name_or_id ... | vpc_group_id ...] [[--filter name=value] ...]`

ec2-authorize (ec2auth)
Adds a rule to an EC2 security group.
`ec2-authorize ec2_group_name_or_id`

Tagging Tools

ec2-create-tags (ec2addtag)
Adds or overwrites one or more tags for the specified resource or resources. Each tag consists of a key and an optional value. Tag keys must be unique per resource.
`ec2-create-tags resource_id [resource_id ...] --tag key[=value] [--tag key[=value] ...]`

ec2-delete-tags (ec2deltag)
Removes a set of tags from a set of resources. The tag value is not required.
`ec2-delete-tags resource_id [resource_id ...] --tag key[=value] [--tag key[=value] ...]`

ec2-describe-tags (ec2dtag)
Lists your tags. You can filter the list to return only tags you specify.
`ec2-describe-tags [[--filter name=value] ...]`

Other Tools

ec2-get-console-output (ec2gcons)
Retrieves console output for the specified instance.
`ec2-get-console-output instance_id [--raw-console-output]`

This *Amazon Elastic Compute Cloud Quick Reference Card* contains commonly used commands and options. For complete reference information, see the Amazon EC2 Command Line Reference at http://aws.amazon.com/documentation/ec2/.

Platform as a Service (PaaS)

- **Programming Environment**
- **Execution Environment**

- The consumer controls the applications that run in the environment (and possibly has some control over the hosting environment), but does not control the operating system, hardware or network infrastructure on which they are running.

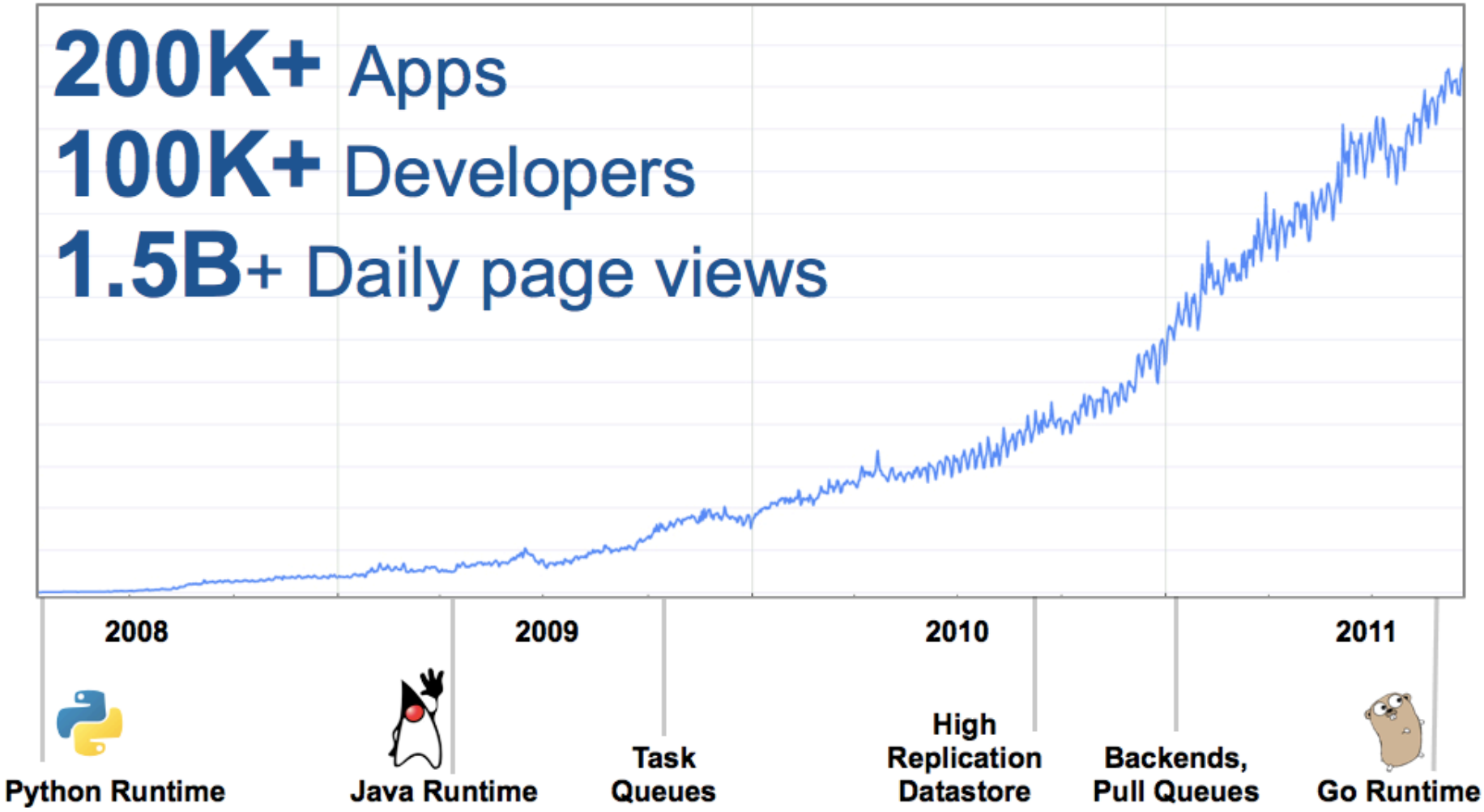
- The platform is typically an application framework.

- Examples:
  - Microsoft Windows Azure (http://www.windowsazure.com)
  - Google App Engine (https://developers.google.com/appengine/)

What is Google App Engine ?

- **Platform as a Service**
- **Scalable serving infrastructure**
  - Build web services to run on Google's infrastructure
  - CPU and data store
- **Software Development Kit with Python, Java and Go runtime**
  - Let developers be developers, not sysadmins
  - Forget about maintaining your own machines - Google holds the pager so you don't have to
- **Web based admin console**
  - Run web analytics
  - Pay only for what you use

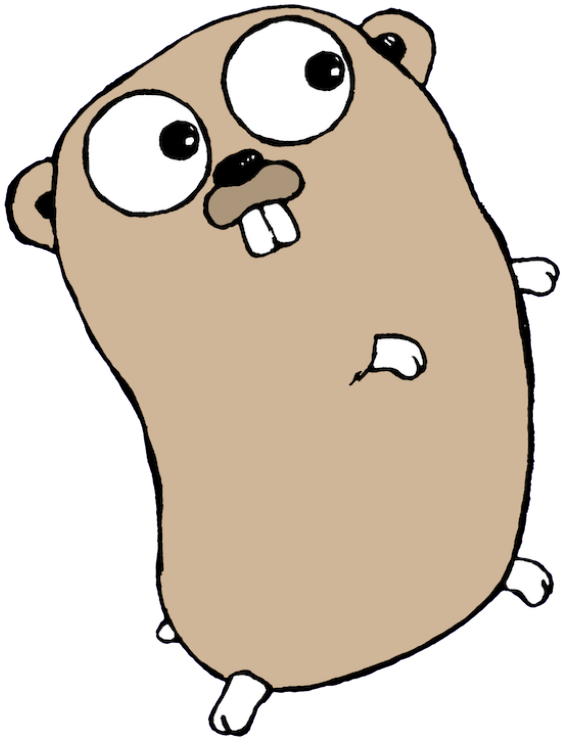
App Engine History



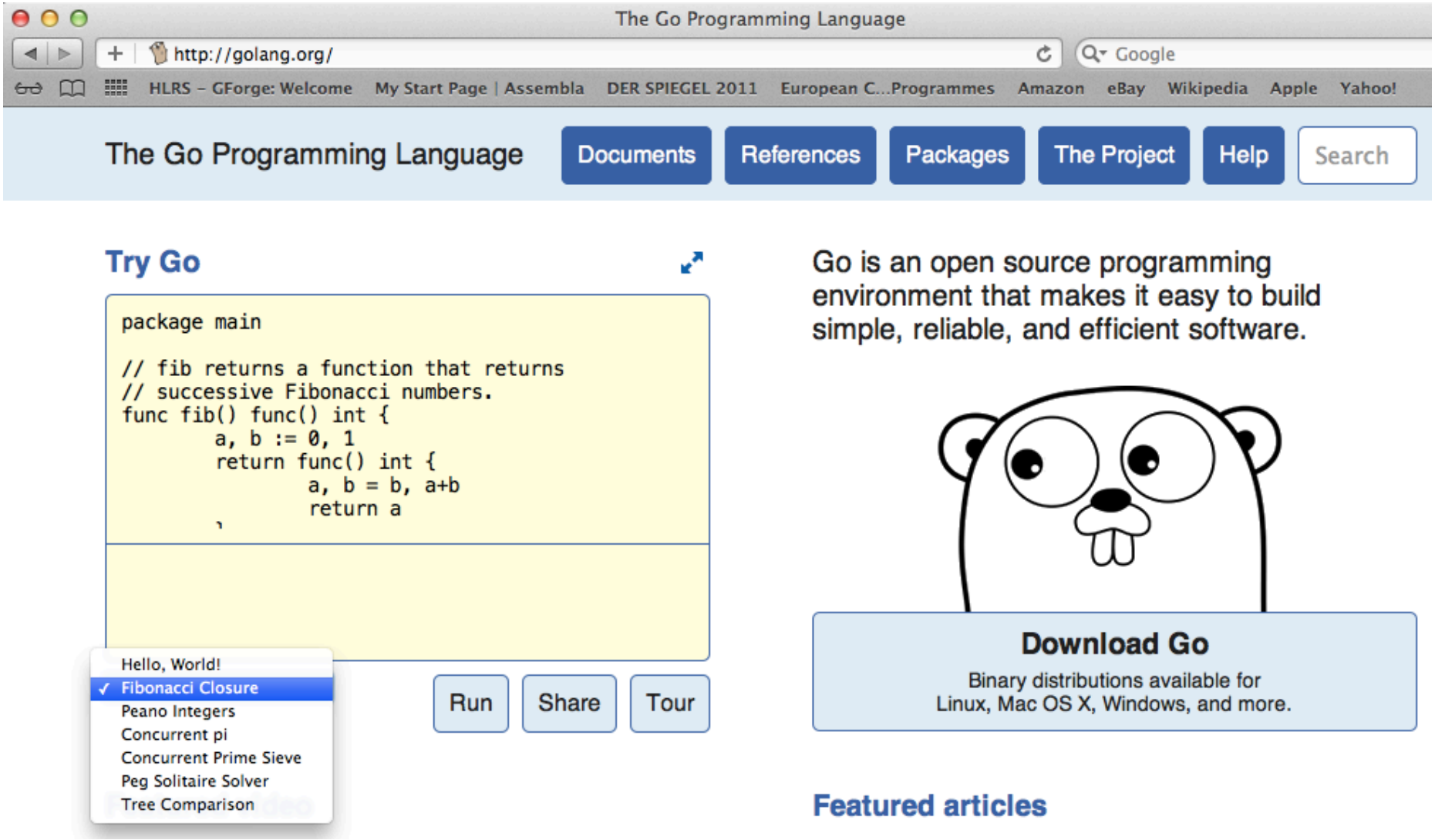
The Go Programming Language

- **Go is a modern, general-purpose language**
  - Compiles to native machine code
  - Statically typed
  - Lightweight syntax
  - Simple type system
  - Concurrency primitives
  - Cool mascot (the gopher)

- **A "canonical language" at Google**



Go Language


The Go Programming Language
http://golang.org/
Google
HLRS – GForge: Welcome
My Start Page | Assembla
DER SPIEGEL 2011
European C...Programmes
Amazon
eBay
Wikipedia
Apple
Yahoo!
The Go Programming Language
Documents
References
Packages
The Project
Help
Search
Try Go
package main
// fib returns a function that returns
// successive Fibonacci numbers.
func fib() func() int {
a, b := 0, 1
return func() int {
a, b = b, a+b
return a
Hello, World!
Fibonacci Closure
Peano Integers
Concurrent pi
Concurrent Prime Sieve
Peg Solitaire Solver
Tree Comparison
Run
Share
Tour
Go is an open source programming environment that makes it easy to build simple, reliable, and efficient software.
Download Go
Binary distributions available for Linux, Mac OS X, Windows, and more.
Featured articles

Get started with Go

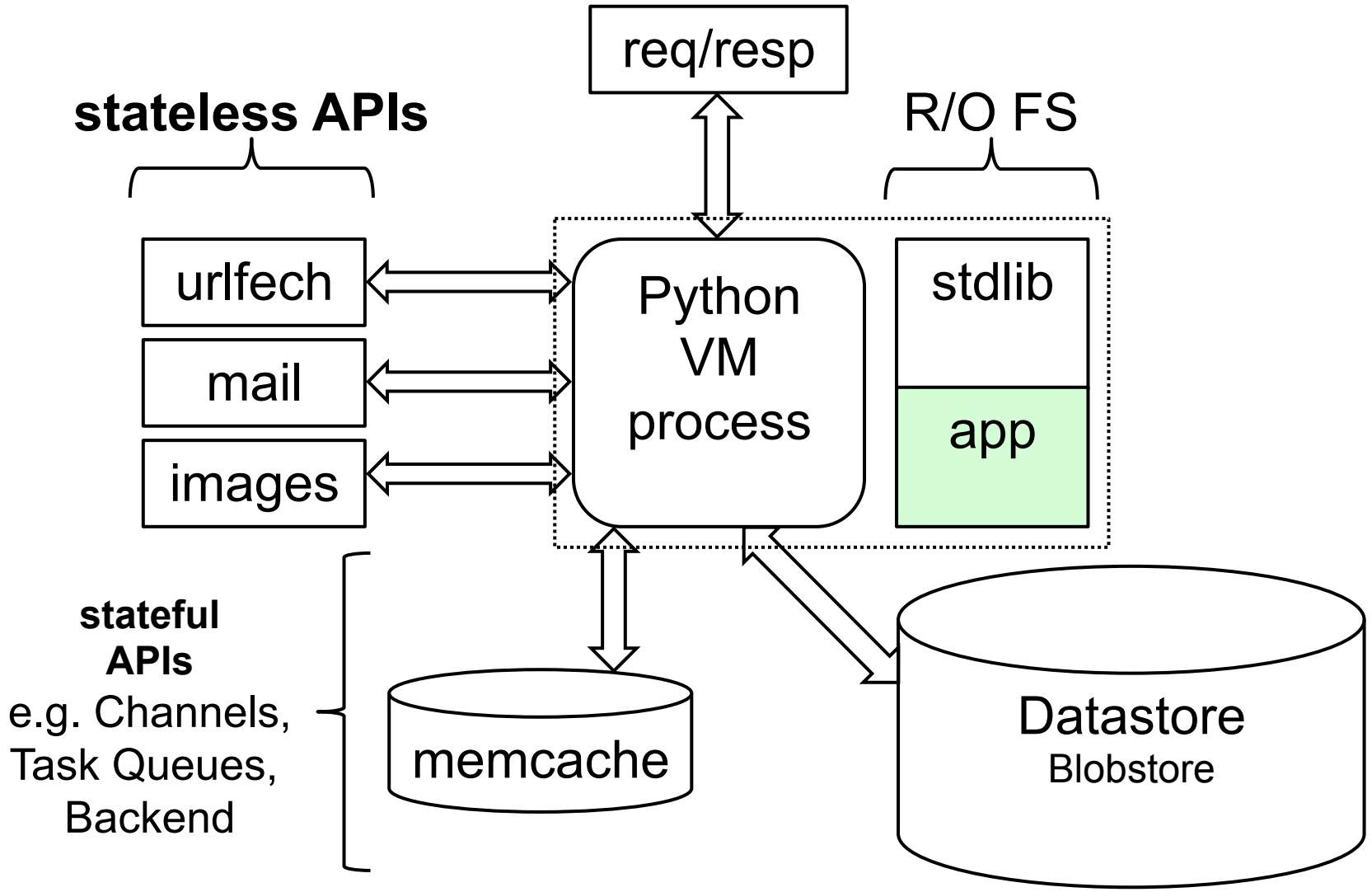
- **The Go Programming Language**
  - http://golang.org/ - homepage
  - http://blog.golang.org/ - official blog
- **A tour of Go - an interactive Go tutorial**
  - http://tour.golang.org/
- **Go for App Engine**
  - http://code.google.com/appengine/docs/go/

App Engine Services

- Datastore - structured, indexed data store
- Blobstore - unstructured key/value store
- Channel - push data from the server to the browser
- Task Queues - process jobs asynchronously
- Backends - long-running processes

- and many more!

App Engine Architecture



Automatic Scaling to Application Needs

- **You don't need to configure your resource needs**
- **One CPU can handle many requests per second**
- **Apps are hashed (really mapped) onto CPUs:**
  - One process per app, many apps per CPU
  - Creating a new process is a matter of cloning a generic "model" process and then loading the application code (in fact the clones are pre-created and sit in a queue)
  - The process hangs around to handle more requests (reuse)
  - Eventually old processes are killed (recycle)
- **Busy apps (many QPS) get assigned to multiple CPUs**
  - This automatically adapts to the need
    - as long as CPUs are available

Backends

- **App instances without request deadlines (Long-running processes)**
- **Varying instance sizes for different workloads (RAM+CPU)**

B1	128MB	600MHz
B2 (default)	256MB	1.2GHz
B4	512MB	2.4GHz
B8	1024MB	4.8GHz

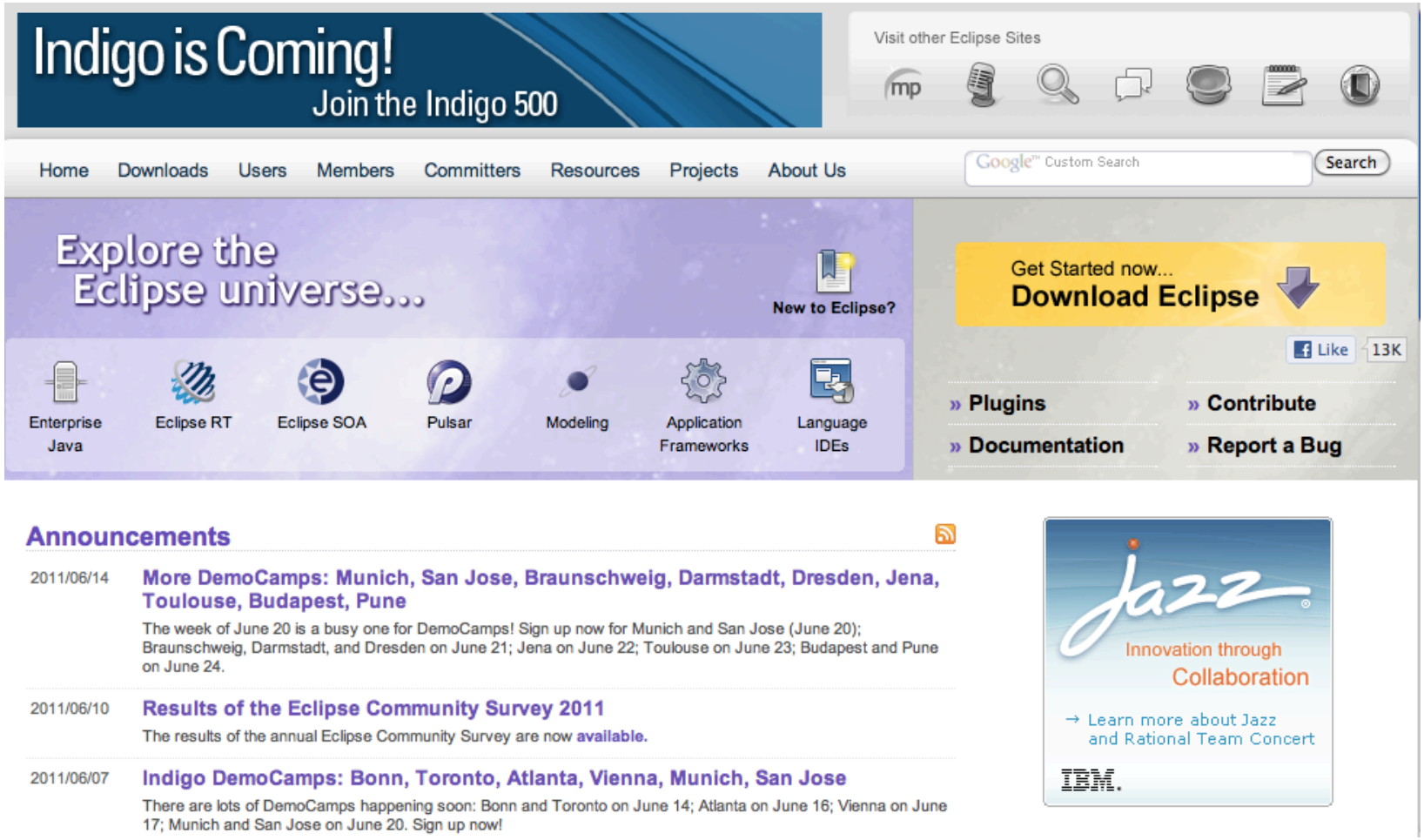
- **Example uses:**
  - Memory-intensive tasks (Search index, game state, cache,…)
  - Background processing (Report generation, indexing, data grooming,…)
  - CPU-intensive tasks (Image manipulation, transcoding, scientific computing, …)

Billable Quota Unit Cost

Resource	Unit	Unit cost
Outgoing Bandwidth	gigabytes	$0.12
Incoming Bandwidth	gigabytes	$0.10
CPU Time	CPU hours	$0.10
Stored Data	gigabytes per month	$0.15
High Replication Storage	gigabytes per month	$0.45
Recipients Emailed	recipients	$0.0001
Always On	N/A (daily)	$0.30
Backends (`B1` class)	Hourly per instance	$0.08
Backends (`B2` class)	Hourly per instance	$0.16
Backends (`B4` class)	Hourly per instance	$0.32
Backends (`B8` class)	Hourly per instance	$0.64

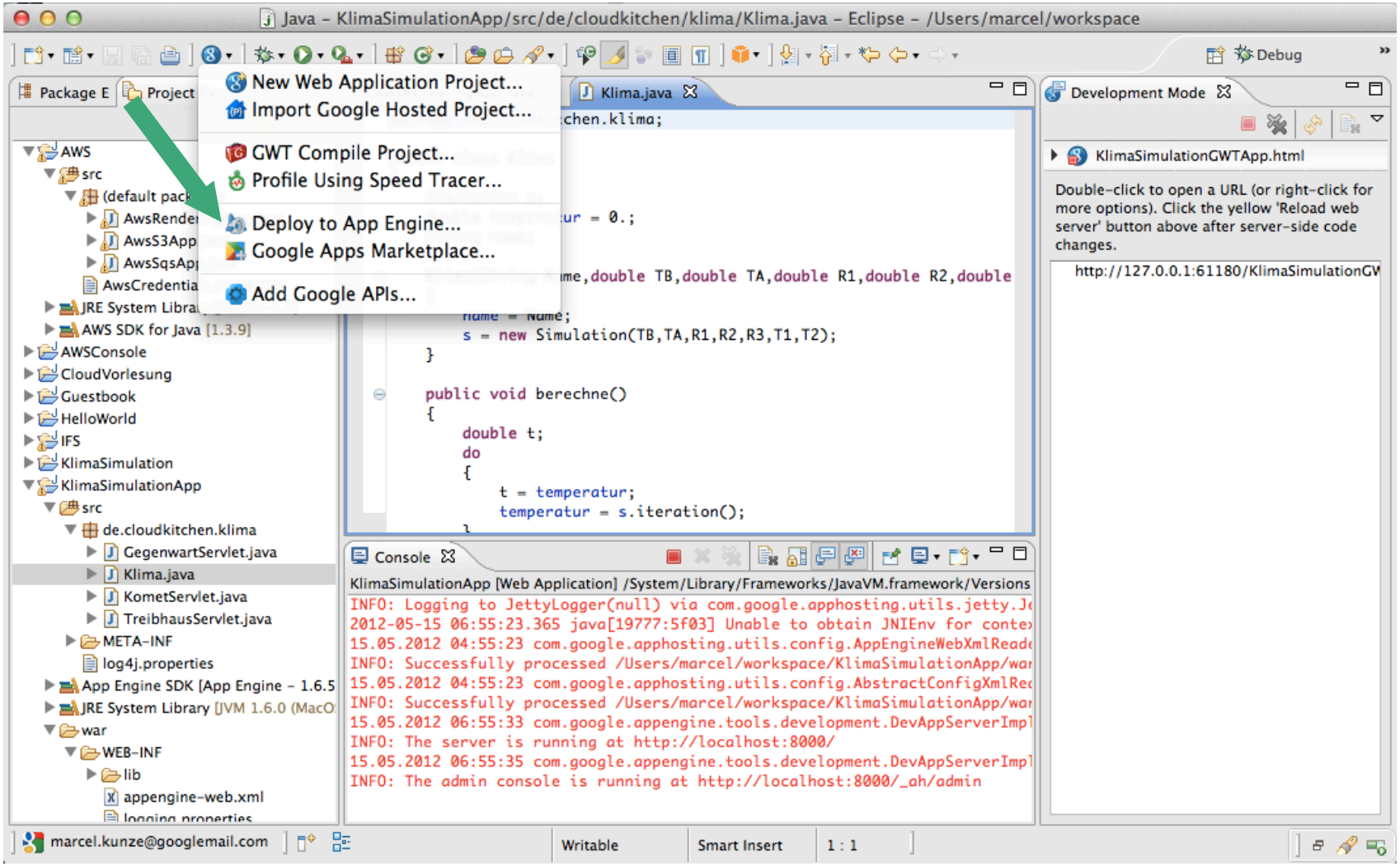
- **Free quota usage corresponds to approx. 5 million page hits per month**
- **Max. daily budget can be set**
- **Backend services for long lasting jobs**

Software Development Tools



- IDE = Integrated Development Environment
- Eclipse framework is very prominent: http://www.eclipse.org/

GAE Plugin for Eclipse IDE / SDK



https://developers.google.com/eclipse/

6. Applications

How to build scalable Applications ?

- Rule 1: Service all web requests
- Rule 2: Service requests as fast as possible
- Rule 3: Handle requests at any scale
- Rule 4: Simplify architecture with services
- Rule 5: Automate operational management
- Rule 6: Leverage unique cloud properties

Source: aws.amazon.com

Best Practices I

1) **Build the application as a service.** Because you are deploying one or more full virtual machines and because clouds are designed to host web services, you want your application to support multiple users or, at least, a sequence of multiple executions.
   - If you are not using the application, scale down the number of servers and scale up with demand.
   - Attempting to deploy 100 VMs to run a program that executes for 10 minutes is a waste of resources because the deployment may take more than 10 minutes.
   - To minimize start up time one needs to have services running continuously ready to process the incoming demand.
2) **Build on existing cloud deployments.** For example use an existing MapReduce deployment such as Hadoop or existing Roles and Appliances (Images)

Source: G.Fox, IU

Best Practices II

3) **Use PaaS if possible.** For Platform-as-a-Service clouds use the tools that are provided such as queues, and blob, table and SQL storage.

4) **Design for failure.** Applications that are services running forever will experience failures. The cloud has mechanisms to automatically recover lost resources, but the application needs to be designed to be fault tolerant.

   - In particular, environments like MapReduce/Hadoop will automatically recover many explicit failures and adopt scheduling strategies that recover performance "failures" from for example delayed tasks.
   - One expects an increasing number of such platform features to be offered by clouds and users will still need to program in a fashion that allows task failures but be rewarded by environments that transparently cope with these failures.

Source: G.Fox, IU

Best Practices III

5) **Use "X as a Service" where possible.** Capabilities such as SQLaaS (database as a service or a database appliance) provide a friendlier approach than the traditional non-cloud approach exemplified by installing MySQL on the local disk.

6) **Moving Data is a challenge.** The general rule is that one should move computation to the data, but if the only computational resource available is a the cloud, you are stuck if the data is not also there.

Source: G.Fox, IU

1) Transformation of Software Licenses to SaaS

United States

Products & Services | Solutions | Academia | Support | User Community | Events | Company

Products & Services > MATLAB

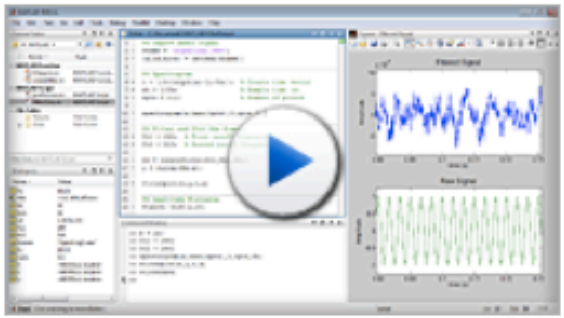
MATLAB
The Language of Technical Computing

Overview

Videos & Examples

Webinars

MATLAB is a programming environment for algorithm development, data analysis, visualization, and numerical computation. Using MATLAB, you can solve technical computing problems faster than with traditional programming languages, such as C, C++, and Fortran.



Product Overview 2:04

You can use MATLAB in a wide range of applications, including signal and image processing, communications, control design, test and measurement, financial modeling and analysis, and computational biology. For a million engineers and scientists in industry and academia, MATLAB is the language of technical computing.

Source: MathWorks

MathWorks | Accelerating the pace of engineering and science

United States | Contact Us

Products & Services | Solutions | Academia | Support | User Community | Events | Company

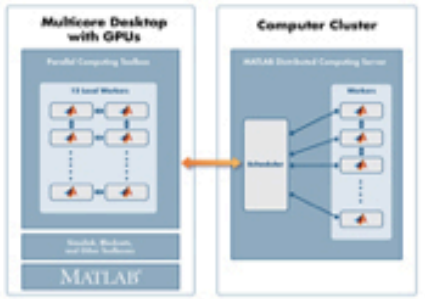
- Product Overview
  - Description
  - Function List
  - Supported Schedulers
  - Examples and Webinars
  - Related Products
  - System Requirements
  - Latest Features
- Support & Training
  - Product Support
  - Documentation
  - Installation Instructions
  - Downloads & Trials
- Other Resources
  - Technical Literature
  - User Stories

MATLAB Distributed Computing Server

MAJOR UPDATE

Perform MATLAB and Simulink computations on clusters, clouds, and grids

MATLAB Distributed Computing Server™ lets you run computationally intensive MATLAB® programs and Simulink® models on computer clusters, clouds, and grids. You develop your program or model on a multicore desktop computer using Parallel Computing Toolbox™ and then scale up to many computers by running it on MATLAB Distributed Computing Server. The server supports batch jobs, parallel computations, and distributed large data. The server includes a built-in cluster job scheduler and provides support for commonly used third-party schedulers.



MATLAB Distributed Computing Server provides licenses for all MathWorks toolboxes and blocksets, so you can run your MATLAB programs on a cluster without having to separately acquire additional product-specific licenses for each computer in the cluster.

See the list of products ineligible for use with MATLAB Distributed Computing Server.

Source: MathWorks

MathWorks | Accelerating the pace of engineering and science

United States | Contact Us | Store

Create Account | Log In

Products & Services | Solutions | Academia | Support | User Community | Events | Company

MATLAB® Mobile™

Connect to MATLAB remotely from your iPhone, iPad, or iPod touch.

Overview

Connect to the Cloud

Connect to Your Computer

Videos and Examples

System Requirements

FAQ

Connect to the Cloud

Connecting to the MathWorks Cloud provides access to a MATLAB session wherever you have Internet access from your iOS device. With a cloud connection, you can perform simple calculations and prototyping when your computer is not accessible.

Don't see the download buttons?

Log in to your MathWorks Account or create an account now.

Getting Started

To set up a MATLAB Mobile connection to the MathWorks Cloud:

1. **On your computer, verify that you have a MathWorks Account.**
   - Log in to your MathWorks Account or create an account now.
2. **Verify that you have an up-to-date license associated with your account.**
   - Go to License Center and check that your account is associated with a license.
   - If you do not have an associated license, click the Add License button and follow the directions.

MATLAB Mobile is hosted on Amazon Web Services !

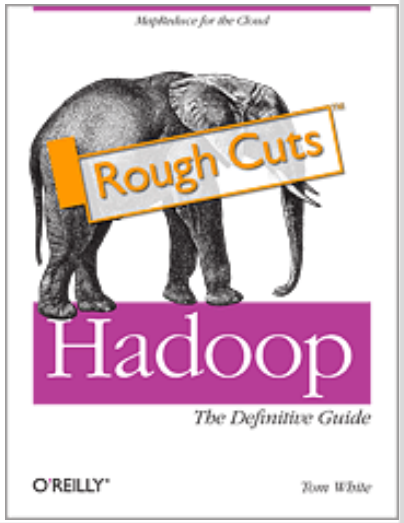
Source: MathWorks

2) Big Data Analysis

http://www.cloudera.com/

What Is hadoop ?

- **Distributed computing frame work**
  - For clusters of computers
  - Thousands of compute nodes
  - Petabytes of data
- **Open source, Java**
- **Now part of Apache group**
- **Reproduce the proprietary MapReduce software infrastructure developed by Google**



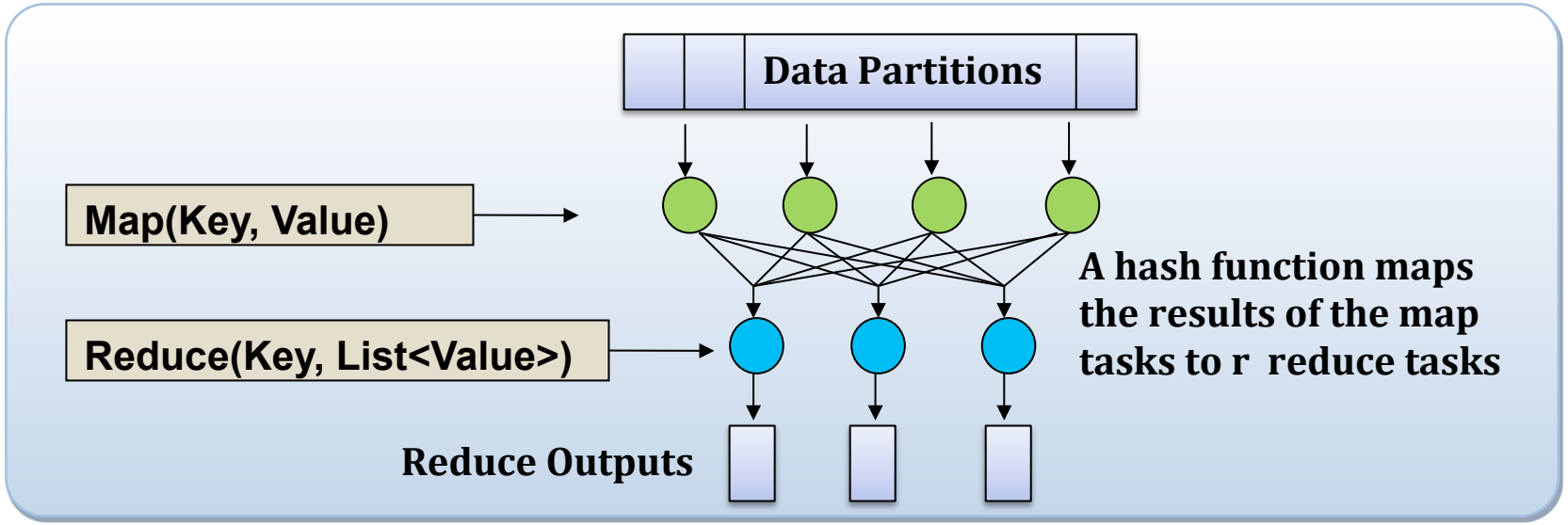
Motivation: Large Scale Data Processing

- **MapReduce: Algorithm for large scale parallel data processing**
  - Want to process lots of data (> 1 TB)
  - Want to parallelize across hundreds/thousands of CPUs
  - … Want to make this easy

- **Potential fields of application**
  - Web indexing
  - Data mining
  - Log file analysis
  - Machine learning
  - Scientific simulation
  - Bioinformatics research

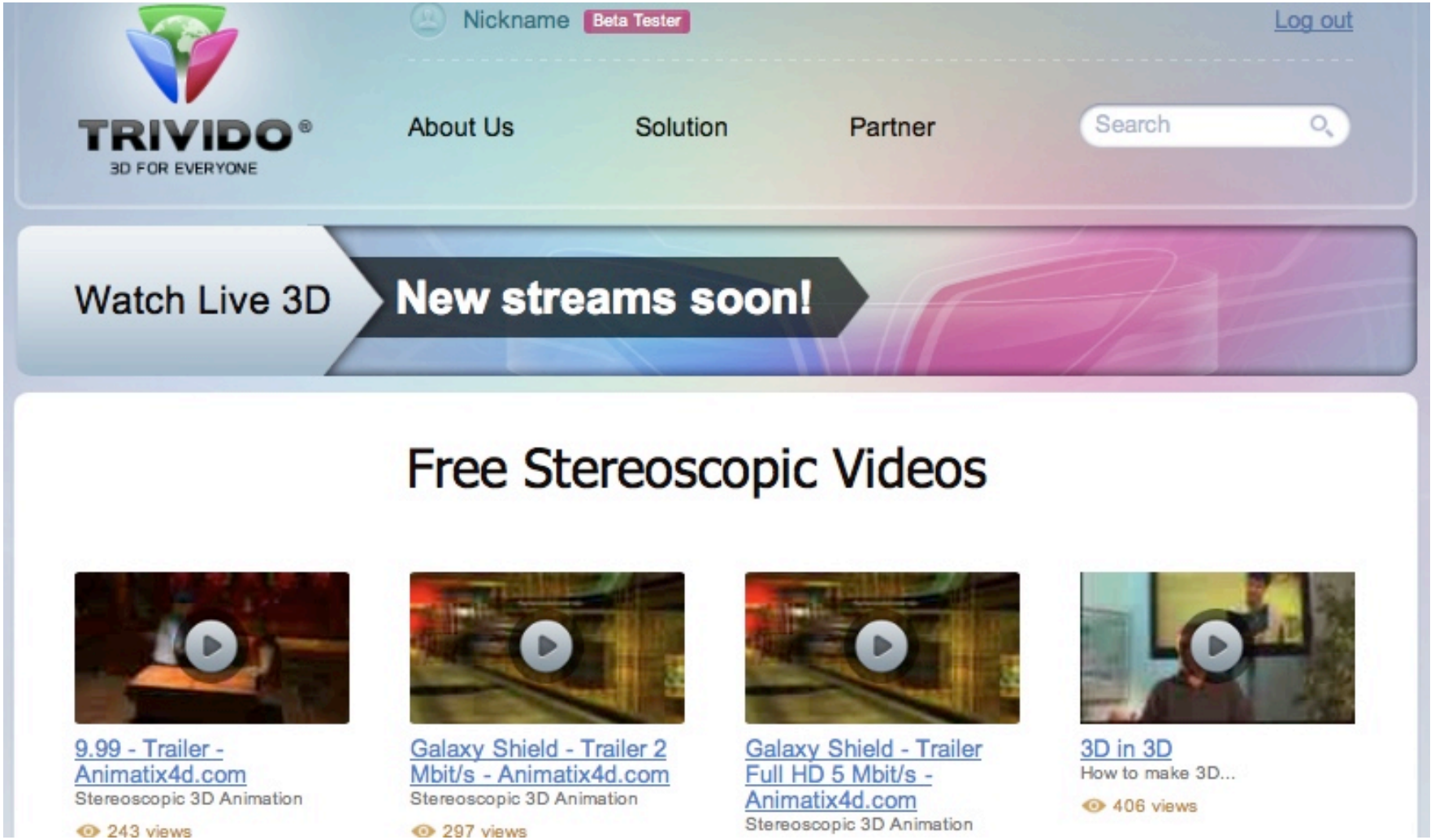
- **Google uses MapReduce everywhere, e.g. to run PageRank**
- **MapReduce adheres to functional programming paradigm**

MapReduce



- **Implementations support:**
  - Splitting of data
  - Passing the output of map functions to reduce functions
  - Sorting the inputs to the reduce function based on the intermediate keys
  - Quality of service
- **AWS offers an "Elastic MapReduce" cloud service that may be consumed by the hour**

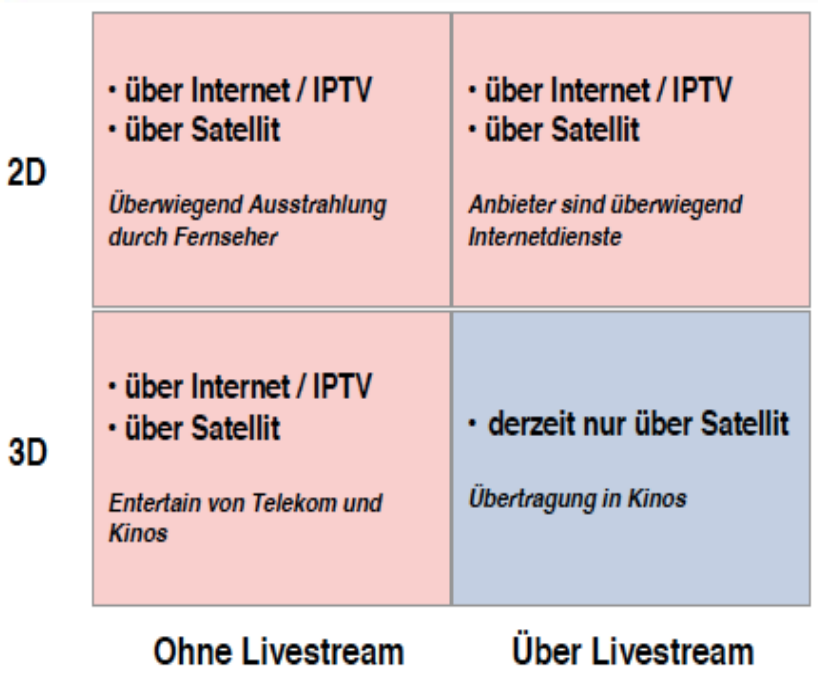
3) Stereoscopic 3D Live Streaming (S3D)



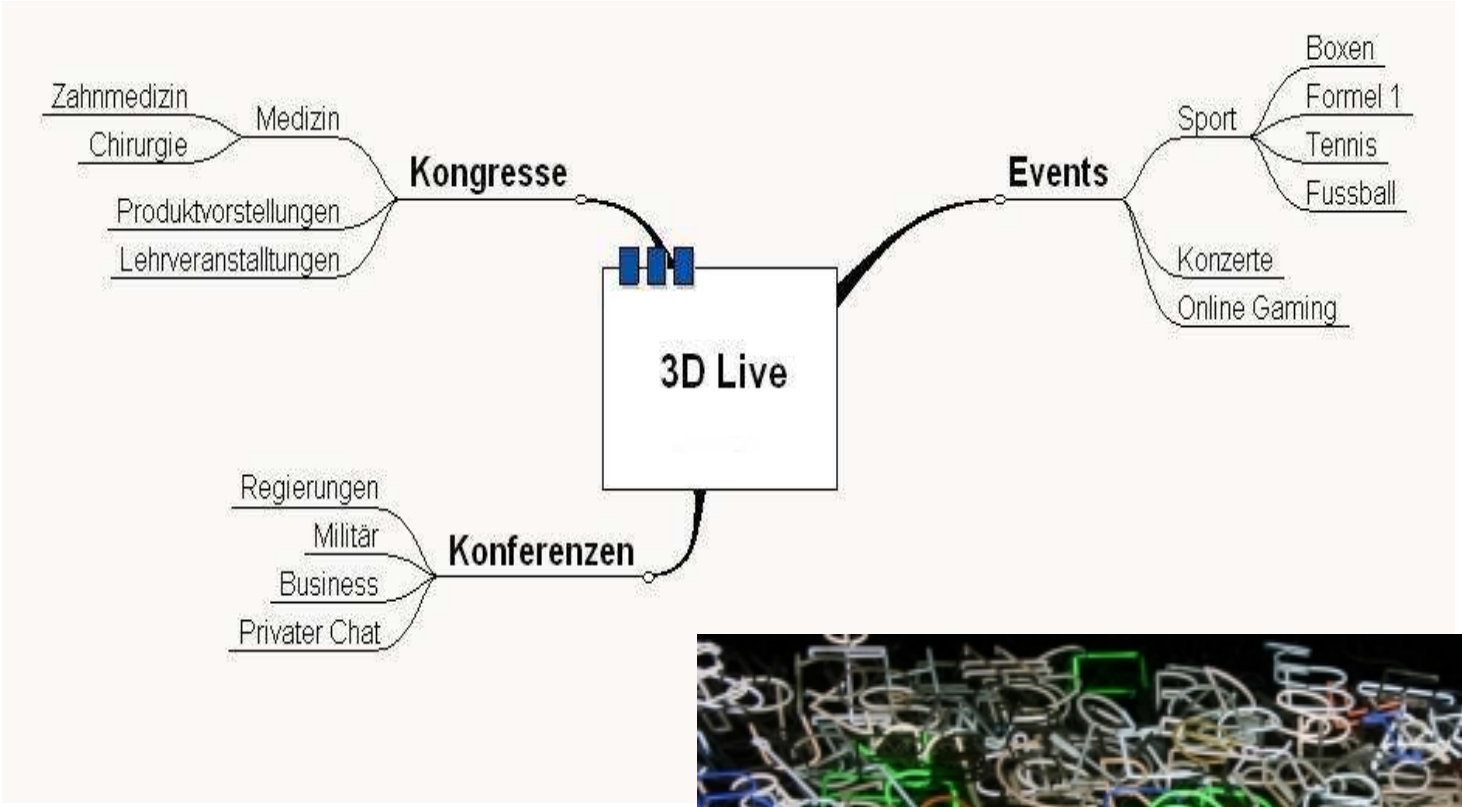
Source: www.trivido.com

S3D Live Streaming: Cost Consideration

- **Live streaming is traditionally done via Satellite**
  - Expensive: Equipment $2000/d, satellite $700/h
  - Point-to-point only
  - Endpoint: Cinema
  - Needs to be organized far ahead
- **Internet streaming**
  - Affordable: Only needs bandwidth (HD: 2Mbps-40Mbps)
  - Available everywhere: Millions of simultaneous viewers
  - Endpoint: Web browser
  - May be used as a utility on-demand

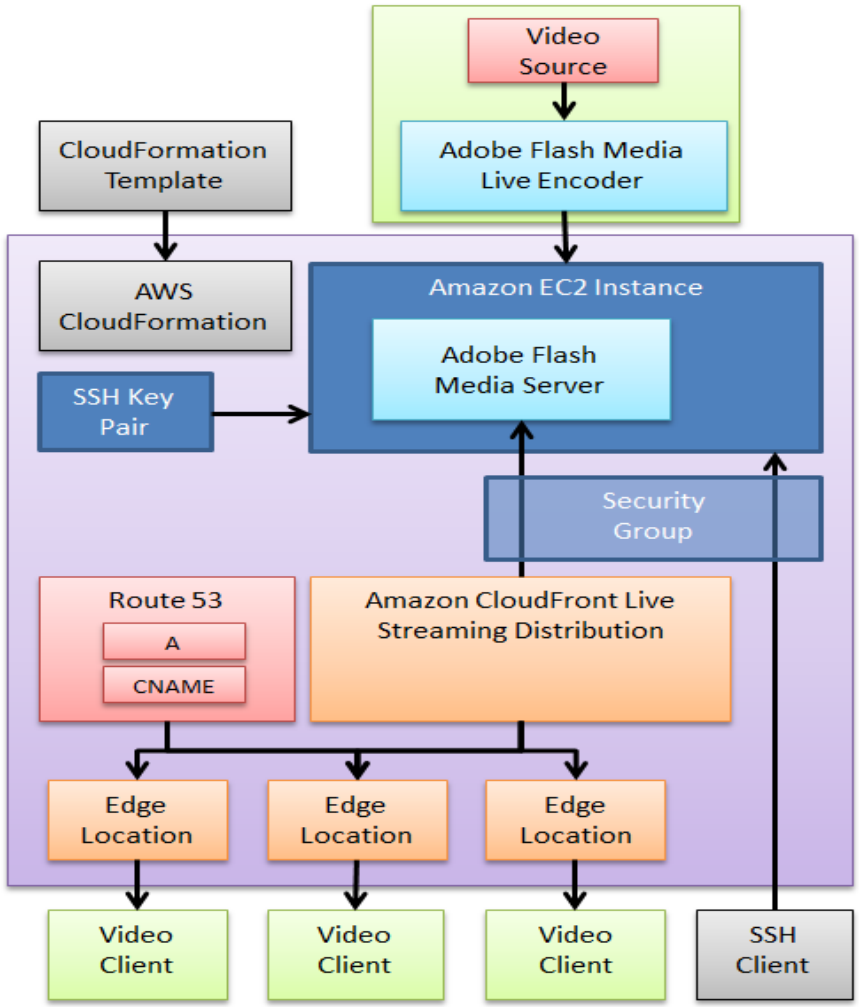


S3D Live Streaming: Application Area



- **Interesting new model to support any kind of live event / lectures**

S3D Live Streaming over the Internet



Source: highedwebtech.com

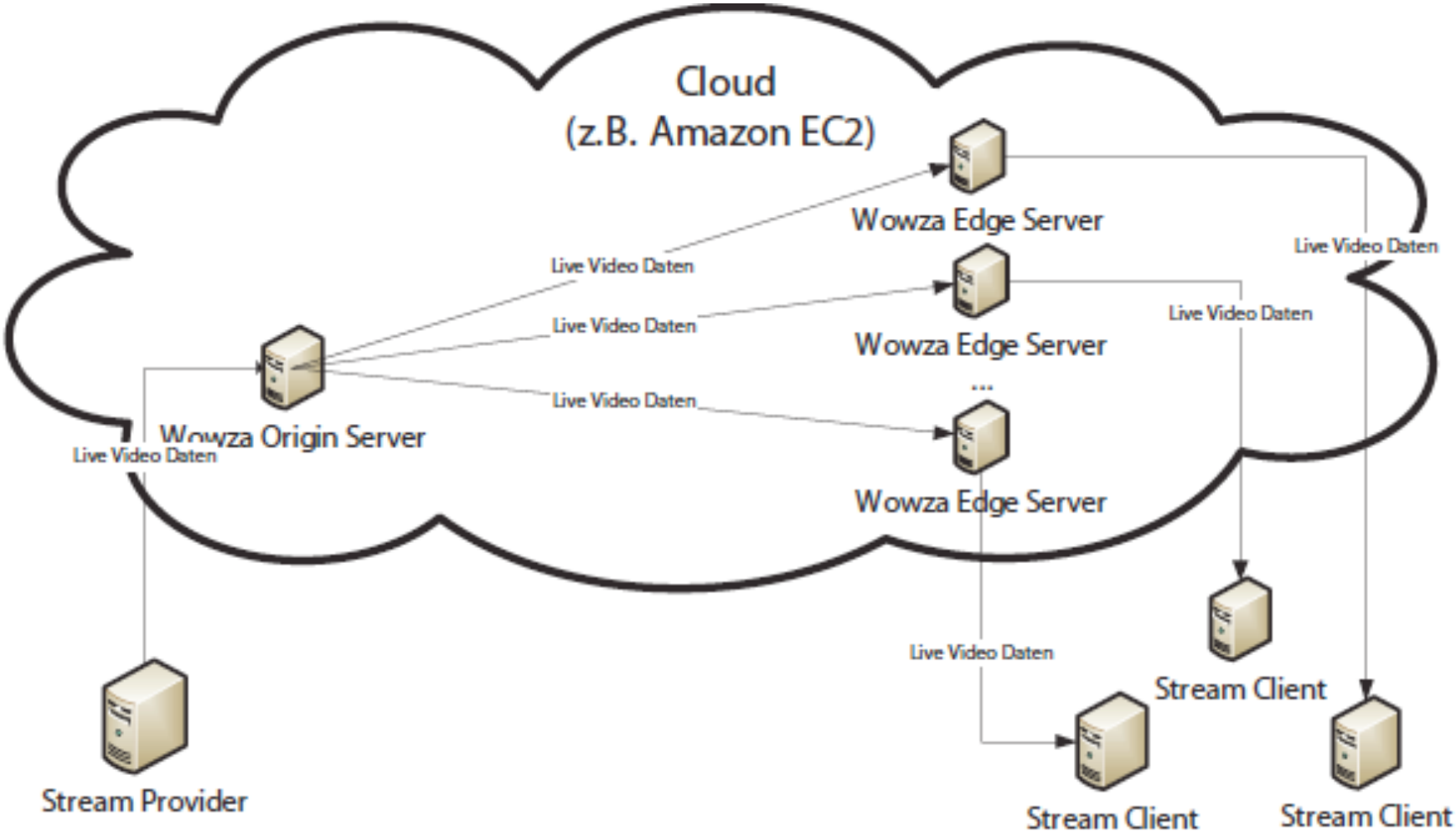
- **Idea: Use Amazon Web Services to transmit 3D stream to 1000's of screens worldwide**
- **Use Adobe Flash streaming services (Wowza, Adobe FMS)**
- **Adobe FMS can be installed as a scalable application by use of the CloudFormation service**
- **Use CloudFront service to distribute Live streams to edge locations**
- **Route53 is a scalable DNS service to manage the IP addresses in a specific domain (e.g. cloudkitchen.de)**
- **NB: NASA is deploying a similar system to support the Curiosity mission: "Mars Science Lab"**


NASA TV
Route 53 DN
Multi-region & A
Elastic Loadbalancin
Amazon Live Streamin
Cloud Fron
Cloud Formatio

S3D Live Streaming: Cloud Service

- **Service to stream stereoscopic 3D content over the Internet**
  - Similar to YouTube
  - Transmission and recording of Live events
  - Transmission and storage of static content
  - A cooperation of Invistra and KIT, see www.trivido.com

S3D Live Streaming: Trivido Architecture

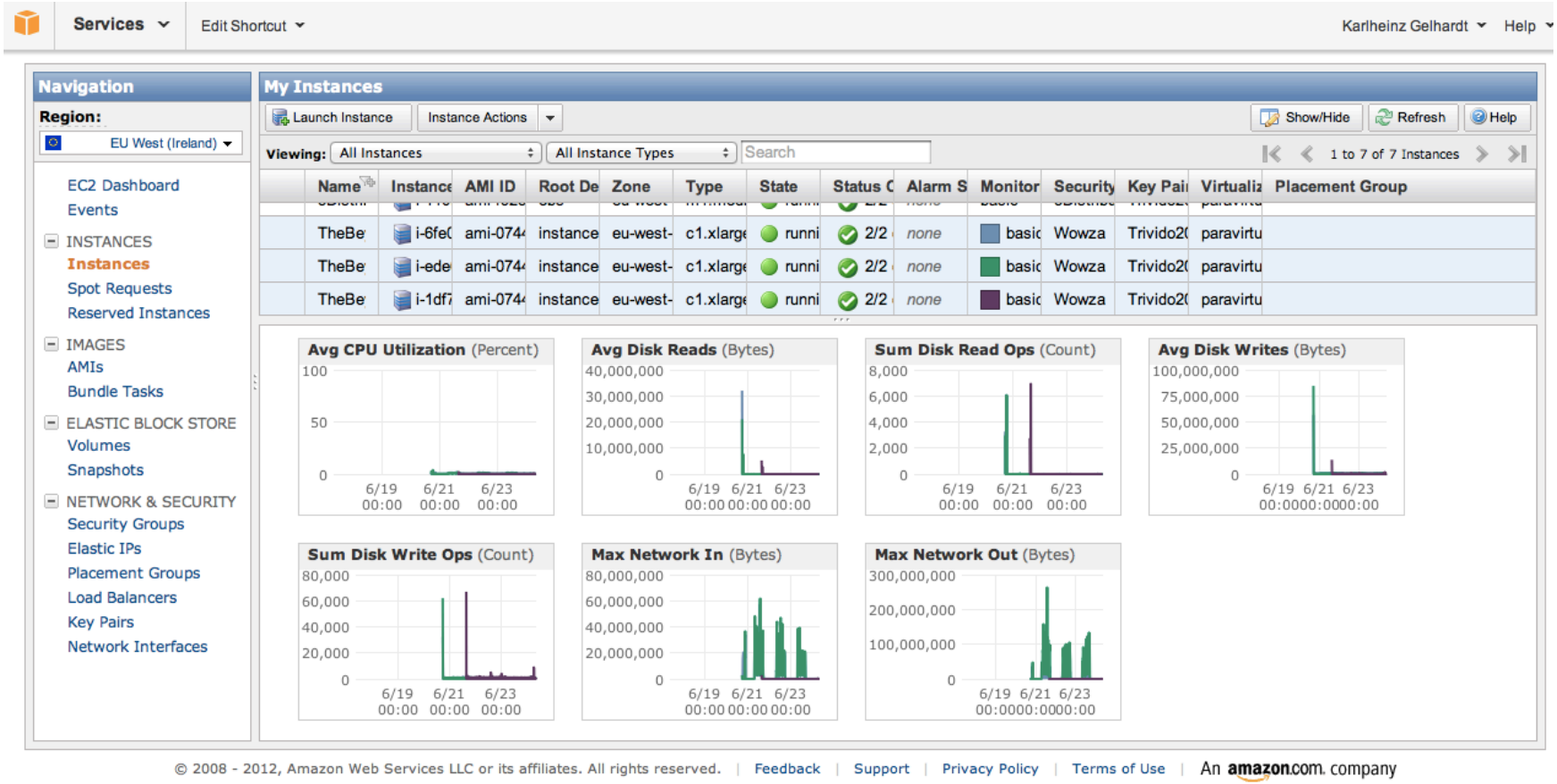


Flash Browser Plugin

S3D Live Streaming: Challenges

- **Implement a distributed system to cover geography and demand**
- **Streams have to be compressed in the multiplexer**
  - 2 Mbps for 720p
  - 5 Mbps for 1080p
  - Medical applications: up to 40 Mbps / 4k resolution
- **Endpoint has to support a variety of formats (side-by-side, anaglyph,…) and devices (3D TV, Polarization, Shutter,…)**
- **Requirement: Secure the streams and contents in a way that they could not be recorded at the endpoint**

S3D Live Streaming: Results



- **It is possible to feed thousands of streams simultaneously**
  - Streaming produces substantial network I/O
  - Almost no CPU as we do not transcode movies (e.g. Cupertino streams for iPhone)

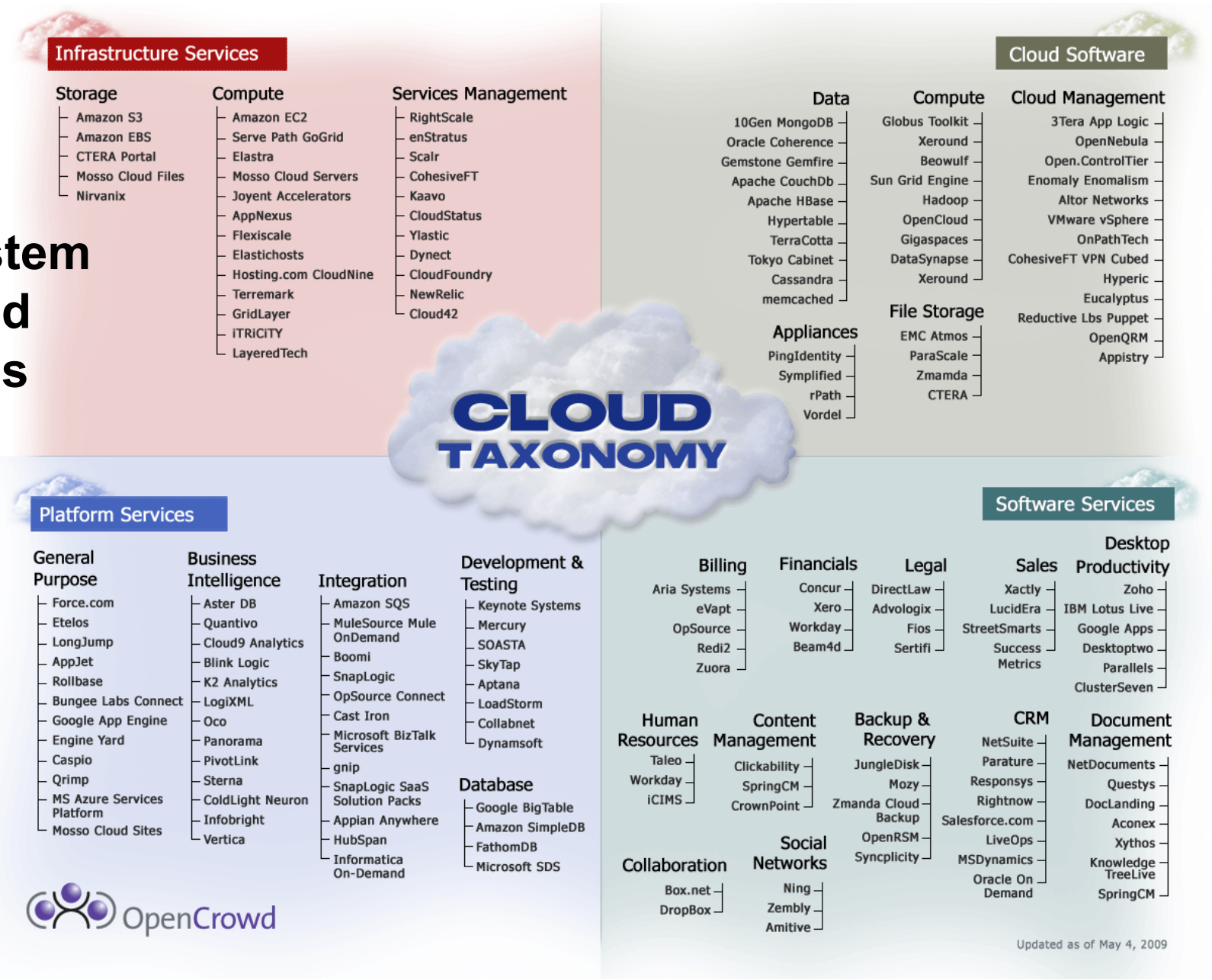
Summary and Outlook

- **The world is moving towards the IT service supermarket**
  - Self service
  - Standardization of requests (small, medium, large, XL, …)
  - Automation of service lifecycle w/o maintenance windows
  - Business models
- **Multi tenancy is key to gain trust**
  - Tenants are privileged
  - Tenants are independent of each other
  - Tenants always get what they want immediately
  - Tenants pay as they go
- **Democratization of IT is empowering the employees**
  - Knowledge workers are able to service themselves
  - Knowledge workers are able to make a suiting choice in the IT market

Offerings in the Public Cloud

- **Lots of interesting commercial services exist**
  - Amazon Web Services, Google Storage, etc. (IaaS)
  - Google App Engine, Microsoft Azure, etc. (PaaS)
  - Google Apps, Microsoft Office Live, Salesforce, etc. (SaaS)
- **Some services are free to a certain extent**
  - Google Apps, Google App Engine, Google Storage, Apple iCloud, …
  - Assembla subversion services (Code management)
    http://www.assembla.com/
  - DropBox, Google Drive etc. (Cloud “USB-stick”, replaces home directory), works with any platform, also with smartphones, iPad etc.
    http://www.dropbox.com/, https://drive.google.com
- **Opportunities**
  - Amazon teaching grants and research grants (quarterly call for proposals)
    http://aws.amazon.com/education/

Ecosystem of Cloud services



Business Models in the Cloud

- **Use of credit card for authentication and billing**
  - An employee might use his own credit card (Happens very often!)
  - An employee might be allowed to use an enterprise credit card
  - An employee might claim Internet services as travel expenses
- **An enterprise might want to establish an IT service center rather than a computing center**
  - **Importance of Service Request Management that is independent of operational units (“System Services” rather than “Systems & Servers”)**
  - Organizes the market place
  - Integrates cloud services into the processes
  - Takes care of capacity planning and monitoring
  - Negotiates service delivery with the cloud providers
  - Takes care of billing
    - E.g. buy Amazon Web Services vouchers and hand them out to employees
- **Traditional computing centers in the enterprise will have to change!**

Cloud Computing is just a new Paradigm...

- 70's: Mainframe
- 80's: PC
- 90's: Workstation
- 00's: Grid
- 10's: Cloud

… But it will change the world like no other disruptive computing technology before !

From http://blogs.zdnet.com/Hinchcliffe


©johnlund.com

Acknowledgements

- Stefan Tai, KIT
- Maximilian Hoecker, KIT