Cloud Computing mit mathematischen Anwendungen

Dr. habil. Marcel Kunze
Engineering Mathematics and Computing Lab (EMCL)
Institut für Angewandte und Numerische Mathematik IV
Karlsruhe Institute of Technology (KIT)
Lectures

VL01 16.04. Intro
VL02 23.04. Basics
VL03 30.04. IaaS, PaaS
VL04 07.05. SaaS, Cloud Architecture
VL05 14.05. Programming Models
VL06 21.05. Hadoop 1
VL07 04.06. Hadoop 2
VL08 11.06. Parallel Computing 1
VL09 18.06. Parallel Computing 2
VL10 25.06. Applications 1
VL11 02.07. Applications 2
VL12 09.07. Security
VL13 16.07. Summary, Q&A

28.05. No Lecture!
Amazon 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, fast, inexpensive data storage 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.
Amazon S3

- Write, read, and delete objects containing up to 1 terabyte of data each. The number of objects you can store is unlimited.
- Each object is stored in a bucket and retrieved via a unique, developer-assigned key.
A bucket can be located in the United States, in Europe or in Asia. All objects within the bucket will be stored in the bucket’s location, but the objects can be accessed from anywhere.

Authentication mechanisms are provided to ensure that data is kept secure from unauthorized access. Objects can be made private or public, and rights can be granted to specific users.

Uses standards-based REST and SOAP interfaces designed to work with any Internet-development toolkit.

Built to be flexible so that protocol or functional layers can easily be added. Default download protocol is HTTP. A BitTorrent™ protocol interface is provided to lower costs for high-scale distribution. Additional interfaces will be added in the future.

Reliability backed with the Amazon S3 Service Level Agreement.

Source: aws.amazon.com
Amazon S3 Namespace

Amazon S3

bucket

object

object

bucket

object

object

bucket

object

object
## S3 Pricing

### Storage Pricing

<table>
<thead>
<tr>
<th>Region: US Standard</th>
<th>Standard Storage</th>
<th>Reduced Redundancy Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>First 1 TB / month</td>
<td>$0.125 per GB</td>
<td>$0.093 per GB</td>
</tr>
<tr>
<td>Next 49 TB / month</td>
<td>$0.110 per GB</td>
<td>$0.083 per GB</td>
</tr>
<tr>
<td>Next 450 TB / month</td>
<td>$0.095 per GB</td>
<td>$0.073 per GB</td>
</tr>
<tr>
<td>Next 500 TB / month</td>
<td>$0.090 per GB</td>
<td>$0.063 per GB</td>
</tr>
<tr>
<td>Next 4000 TB / month</td>
<td>$0.080 per GB</td>
<td>$0.053 per GB</td>
</tr>
<tr>
<td>Over 5000 TB / month</td>
<td>$0.055 per GB</td>
<td>$0.037 per GB</td>
</tr>
</tbody>
</table>

### Request Pricing

<table>
<thead>
<tr>
<th>Region: US Standard</th>
<th>Pricing</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUT, COPY, POST, or LIST Requests</td>
<td>$0.01 per 1,000 requests</td>
</tr>
<tr>
<td>GET and all other Requests †</td>
<td>$0.01 per 10,000 requests</td>
</tr>
</tbody>
</table>

† No charge for delete requests

Source: aws.amazon.com, 2012
## S3 Pricing

### Data Transfer Pricing

<table>
<thead>
<tr>
<th>Region: US Standard</th>
<th>Pricing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Transfer IN</strong></td>
<td></td>
</tr>
<tr>
<td>All data transfer in</td>
<td>$0.000 per GB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Data Transfer OUT</strong></th>
<th>Pricing</th>
</tr>
</thead>
<tbody>
<tr>
<td>First 1 GB / month</td>
<td>$0.000 per GB</td>
</tr>
<tr>
<td>Up to 10 TB / month</td>
<td>$0.120 per GB</td>
</tr>
<tr>
<td>Next 40 TB / month</td>
<td>$0.090 per GB</td>
</tr>
<tr>
<td>Next 100 TB / month</td>
<td>$0.070 per GB</td>
</tr>
<tr>
<td>Next 350 TB / month</td>
<td>$0.050 per GB</td>
</tr>
<tr>
<td>Next 524 TB / month</td>
<td>Contact Us</td>
</tr>
<tr>
<td>Next 4 PB / month</td>
<td>Contact Us</td>
</tr>
<tr>
<td>Greater than 5 PB / month</td>
<td>Contact Us</td>
</tr>
</tbody>
</table>

Source: aws.amazon.com, 2012
Google Cloud Storage

Google Cloud Storage – Store your data in Google’s cloud

Google Cloud Storage makes it easy for companies to store and reliably access vast amounts of business data.

Highly scalable
Store and manage an unlimited number of objects – up to five terabytes each.

Flexible security and sharing
ACLs for individuals or groups simplify authentication and sharing for stored data. OAuth 2.0 support enhances security and adds flexibility.

Fast data access
Google Storage provides quick and easy access to your data around the world, and offers hosting choices in highly optimized data centers in multiple regions.

Reliable storage service
Google’s proven storage and networking infrastructure provides data redundancy for added reliability. Our 99.9% SLA helps ensure that your data is available when you need it.

- API is compatible to S3 (S3 may be regarded a de-facto standard)
- http://www.google.com/enterprise/cloud/storage/
Pricing is almost compatible S3 (a little bit cheaper than S3)
AWS: Elastic Block Store (EBS)

- 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
Elastic IP Addresses

No cost for Elastic IP addresses while in use

- $0.01 per non-attached Elastic IP address per complete hour
- $0.00 per Elastic IP address remap – first 100 remaps / month
- $0.10 per Elastic IP address remap – additional remap / month over 100

- Provides public IP addresses (Visible on the web)
- Dynamic mapping of public IP address to EC2 instance
Amazon Simple Queue Service (SQS)

Amazon Simple Queue Service (Amazon SQS) offers a reliable, highly scalable, hosted queue for storing messages as they travel between computers. By using Amazon SQS, developers can simply move data between distributed components of their applications that perform different tasks, without losing messages or requiring each component to be always available. Amazon SQS makes it easy to build an automated workflow, working in close conjunction with the Amazon Elastic Compute Cloud (Amazon EC2) and the other AWS infrastructure web services.

Amazon SQS works by exposing Amazon's web-scale messaging infrastructure as a web service. Any computer on the Internet can add or read messages without any installed software or special firewall configurations. Components of applications using Amazon SQS can run independently, and do not need to be on the same network, developed with the same technologies, or running at the same time.
Amazon Simple Queue Service (SQS)

- “Message queuing in the Cloud”
  - Basic message queuing model, except: queues are hosted by Amazon, and queues are accessed using Web service protocols

- Simple API
- Platform agnostic
- Basic support for access control and message locking

- Reliability
  - Runs within Amazon's high-availability data centers
  - Messages stored redundantly across multiple servers and locations
- Scalable to millions of messages a day
SQS Functionality

- Developers can create an unlimited number of Amazon SQS queues, each of which can send and receive an unlimited number of messages.
- New messages can be added to a queue at any time. The message body can contain up to 8 KB of text in any format.
- A computer can check a queue at any time for messages waiting to be read.
- A message is “locked” while a computer is processing it, keeping other computers from trying to process it simultaneously. If processing fails, the lock will expire and the message will again be available.
- Messages can be retained in queues for up to 4 days.
- Developers can access Amazon SQS through standards-based SOAP and Query interfaces designed to work with any Internet-development toolkit.

Source: aws.amazon.com
Management of virtual infrastructure

https://console.aws.amazon.com/
Management of EC2 instances
Management of Amazon Machine Images (AMI)
Storage management (Elastic Block Store)
Management of public IP addresses: Elastic IP

- Dynamic allocation to EC2 instances
Management of key pairs (Privileged access via SSH/RDP)
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/](http://www.s3fox.net/)
Access Control Lists (ACL)

- Define access rights: Read, write, fullcontrol
- Add specific users with e-mail or user ID
AWS command line tools are useful for developers and admins
- Difficult to use for average customer
- Download from http://aws.amazon.com/developertools/351
### Instance Tools

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ec2-run-instances (ec2run)</td>
<td>Launches one or more instances of the specified AMI.</td>
</tr>
<tr>
<td>ec2-terminate-instances (ec2kill)</td>
<td>Terminates the specified instance.</td>
</tr>
<tr>
<td>ec2-stop-instances (ec2stop)</td>
<td>Stops an instance (applies only to Amazon EBS-backed instances).</td>
</tr>
<tr>
<td>ec2-start-instances (ec2start)</td>
<td>Starts a stopped instance (applies only to Amazon EBS-backed instances).</td>
</tr>
<tr>
<td>ec2-monitor-instances (ec2min)</td>
<td>Enables monitoring for the specified instance.</td>
</tr>
<tr>
<td>ec2-unmonitor-instances (ec2unmin)</td>
<td>Disables monitoring for the specified instance(s).</td>
</tr>
</tbody>
</table>

### Amazon EBS Tools

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ec2-create-volume (ec2addvol)</td>
<td>Creates an Amazon EBS volume from a snapshot or an empty volume in the size you specify.</td>
</tr>
<tr>
<td>ec2-create-snapshot (ec2addsnap)</td>
<td>Creates a snapshot of an Amazon EBS volume and stores it in Amazon S3.</td>
</tr>
<tr>
<td>ec2-describe-volumes (ec2dvol)</td>
<td>Lists the specified Amazon EBS volumes. If no volume is specified, all your volumes are listed.</td>
</tr>
<tr>
<td>ec2-list-instances (ec2list)</td>
<td>Lists the instances.</td>
</tr>
<tr>
<td>ec2-terminate-instances (ec2kill)</td>
<td>Terminates the specified instance.</td>
</tr>
<tr>
<td>ec2-stop-instances (ec2stop)</td>
<td>Stops an instance.</td>
</tr>
<tr>
<td>ec2-start-instances (ec2start)</td>
<td>Starts a stopped instance.</td>
</tr>
<tr>
<td>ec2-monitor-instances (ec2min)</td>
<td>Enables monitoring for the specified instance.</td>
</tr>
<tr>
<td>ec2-unmonitor-instances (ec2unmin)</td>
<td>Disables monitoring for the specified instance(s).</td>
</tr>
<tr>
<td>ec2-list-volumes (ec2listvol)</td>
<td>Lists the volumes.</td>
</tr>
</tbody>
</table>

### EC2 Elastic IP Address Tools

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ec2-allocate-address (ecallocateaddr)</td>
<td>Acquires an EC2 Elastic IP address for use with your account.</td>
</tr>
<tr>
<td>ec2-describe-addresses (ec2daddr)</td>
<td>Lists both EC2 and VPC Elastic IP addresses assigned to your account.</td>
</tr>
<tr>
<td>ec2-release-address (ec2reladdr)</td>
<td>Releases an EC2 Elastic IP address associated with your account.</td>
</tr>
<tr>
<td>ec2-associate-address (ec2assocaddr)</td>
<td>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.</td>
</tr>
<tr>
<td>ec2-dissociate-address (ec2disaddr)</td>
<td>Disassociates the specified EC2 Elastic IP address from the instance to which it is assigned.</td>
</tr>
</tbody>
</table>

---

http://docs.amazonwebservices.com/AWSEC2/latest/GettingStartedGuide/
### AWS Quick Reference Card (c’td)

<table>
<thead>
<tr>
<th>Image Tools</th>
<th>Key Pair Tools</th>
<th>Tagging Tools</th>
<th>Other Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>ec2-describe-images (ec2dim)</td>
<td>ec2-create-keypair (ec2addkey)</td>
<td>ec2-create-tags (ec2addtag)</td>
<td>ec2-get-console-output (ec2gcons)</td>
</tr>
<tr>
<td>Returns information about AMIs, AKIs, and ARIs. If no parameter is specified, information about all images for which you have launch permission is returned.</td>
<td>Creates a new 2048-bit RSA key pair with the specified name.</td>
<td>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.</td>
<td>Retrieves console output for the specified instance.</td>
</tr>
<tr>
<td>ec2-describe-images [ami_id ...] [-all] [ -o owner ... ] [x user_id] [ [-filter name=value] ... ]</td>
<td>ec2-describe-keypairs (ec2dkeykey)</td>
<td>ec2-delete-tags (ec2deltag)</td>
<td>ec2-get-console-output instance_id [ -raw-console-output ]</td>
</tr>
<tr>
<td>Creates an AMI that uses an Amazon EBS root device from a running or stopped Amazon EBS-backed instance.</td>
<td>Lists the specified key pairs. If no key pair is specified, all your key pairs are listed.</td>
<td>Removes a set of tags from a set of resources. The tag value is not required.</td>
<td>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 <a href="http://aws.amazon.com/documentation/ec2/">http://aws.amazon.com/documentation/ec2/</a>.</td>
</tr>
<tr>
<td>ec2-create-image instance_id --name name [- --description description] [ --no-reboot]</td>
<td>ec2-delete-keypair (ec2delkey)</td>
<td>ec2-description-tags [ --tag key=value ... ]</td>
<td></td>
</tr>
</tbody>
</table>
Hausaufgabe 2: API-Tools

- Anlegen eines .ec2 Ordners im Home-Verzeichnis
- Bei AWS anmelden, Zertifikat generieren und \textit{pk}…\textit{pem} und \textit{cert}…\textit{pem} Dateien nach .ec2 kopieren (x.509)
- Herunterladen der API-Tools nach \texttt{C:\Programme\ec2-api-tools-1.5.3.0}
- Herunterladen und Installieren von Java
- Setzen von Umgebungsvariablen (Systemsteuerung)
  \begin{align*}
  \text{EC2\_HOME} &= \texttt{C:\Programme\ec2-api-tools-1.5.3.0} \\
  \text{EC2\_CERT} &= \texttt{%HOME%\ec2\cert\.pem} \\
  \text{EC2\_PRIVATE\_KEY} &= \texttt{%HOME%\ec2\pk\.pem} \\
  \text{JAVA\_HOME} &= \texttt{C:\Programme\Java\jdk6}
  \end{align*}
- Falls man in einer speziellen Zone arbeiten will (default: us-east-1)
  \texttt{EC2\_URL=\url{https://eu-west-1.ec2.amazonaws.com}}
- Funktionstest: Zeige verfügbare Zonen mit \textit{ec2-describe-availability-zones}

\begin{tabular}{llll}
\text{AVAILABILITYZONE} & \text{us-east-1a} & \text{available} & \text{us-east-1} \\
\text{AVAILABILITYZONE} & \text{us-east-1b} & \text{available} & \text{us-east-1} \\
\text{AVAILABILITYZONE} & \text{us-east-1c} & \text{available} & \text{us-east-1} \\
\text{AVAILABILITYZONE} & \text{us-east-1d} & \text{available} & \text{us-east-1}
\end{tabular}
Hausaufgabe 2 (c‘td)

- Erzeugen eines Schlüsselpaares (Name z.B. mykey)
  \textit{ec2-add-keypair mykey}
  Der zurück gelieferte private Schlüssel muss in einem File gespeichert werden (z.B. mykey.pem) und die Privilegien sollten auf readonly für den Besitzer gesetzt werden.

- Auflisten der verfügbaren Amazon Machine Images (AMI)
  \textit{ec2-describe-images --owner amazon}
  ...
  IMAGE ami-42a2532b amazon/EC2 CentOS 5.5 GPU HVM AMI (Driver 260.19.29) amazon available public x86_64 machine ebs hvm xen
  ...
  IMAGE ami-e565ba8c amazon/amzn-ami-pv-2012.03.1.x86_64-ebs amazon available public x86_64 machine aki-88aa75e1 ebs paravirtual xen
  ...

- Starten einer virtuellen Maschine (Amazon Linux Micro-Instanz)
  \textit{ec2-run-instances ami-e565ba8c -k mykey -t t1.micro}
Hausaufgabe 2 (c‘td)

- Status der virtuellen Maschine anzeigen
  
  `ec2-describe-instances i-2cb98d43`

  
  RESERVATION r-43976421 default INSTANCE i-ba8685dd ami-e565ba8c `ec2-23-21-35-127.compute-1.amazonaws.com`
  ip-10-244-129-195.ec2.internal running vorlesung-us 0 t1.micro 2012-04-29T16:56:11+0000 us-east-1d
  aki-88aa75e1 monitoring-disabled 23.21.35.127 10.244.129.195 ebs paravirtual xen sg-c1c82da8 default
  BLOCKDEVICE /dev/sda1 vol-7ac4b215 2012-04-29T16:56:31.000Z TAG instance i-ba8685dd Name test

- Firewall für SSH öffnen (Port 22) in der default security group
  
  `ec2-authorize default -p 22`

  PERMISSION default ALLOWS tcp 22 22 FROM CIDR 0.0.0.0/0

- Mit SSH einloggen
  
  `ssh -i mykey.pem root@ec2-23-21-35-127.compute-1.amazonaws.com`

- Stoppen der virtuellen Maschine
  
  `ec2-terminate-instances i-ba8685dd`

  INSTANCE i-10a64379 running shutting-down