

Cloud Day

<http://jclouds.org/>

palestrante: Alisson Wilker

e-mail: alisson-wilker.silva@serpro.gov.br

19 de outubro de 2011

What is jclouds

jclouds is an open source library that helps you get started in the cloud and reuse your Java and Clojure development skills.

It provides an API that allows you freedom to use portable abstractions or cloud-specific features.

It supports 30 cloud providers and cloud software stacks, including Amazon, GoGrid, Ninefold, vCloud, OpenStack, and Azure.

Under Apache 2.0 license.

Why should I use jclouds

Simple Interface

Instead of creating new object types, we reuse concepts like maps so that the programming model is familiar. In this way, you can get started without dealing with REST-like apis or WS.

Runtime Portability

We have drivers that allow you to operate in restricted environments like Google App Engine. We have very few required dependencies, so we are unlikely to clash with your app.

Why should I use jclouds

Deal with Web Complexity

Network based computing introduces issues such as transient failures and redirects. We handle this for you.

Unit Testability

Writing tests for cloud endpoints is difficult. We provide you with Stub connections that simulate a cloud without creating network connections. In this way, you can write your unit tests without mocking complexity or the brittleness of remote connections.

Why should I use jclouds

Location

All of our abstractions are location-aware. For example, you can get ISO-3166 codes to tell which country or province a cloud runs in.

Quality

We test every provider with live scenarios before each release. If it doesn't pass, the provider goes into the sandbox.

Available software

Latest version:

1.1.1

Sources available at GitHub:

<https://github.com/jclouds/jclouds>

Samples available at:

<https://github.com/jclouds/jclouds-examples>

Applications that use jclouds

Twitter

uses whirr to setup test cassandra clusters

GoGrid

utilizes jclouds to test development versions of their cloud platform

Nimble

uses jclouds OSGi integration to deploy Nimble OSGi container instances into Public Clouds (Amazon, GoGrid...)

Other

<http://www.jclouds.org/documentation/reference/apps-that-use-jclouds>

jclouds Abstractions

BlobStore

Simplifies dealing with key-value providers such as Amazon S3. For example, BlobStore can give you a simple Map view of a container.

ComputeService

Simplifies the task of managing machines in the cloud. For example, you can use ComputeService to start 5 machines and install your software on them.

Providers supported by BlobStore

- **aws-s3**
- cloudfiles-us
- cloudfiles-uk
- filesystem
- **azureblob**
- atmos (generic)
- synaptic-storage
- scaleup-storage
- cloudfonestorage
- walrus(generic)

- **googlestorage**
- **ninefold-storage**
- scality-rs2 (generic)
- hosteurope-storage
- tiscali-storage
- **eucalyptus-**
partnercloud-s3
- swift (generic)
- transient (in-mem)

Providers supported by ComputeService

- **aws-ec2**
- **gogrid**
- cloudservers-us
- stub (in-memory)
- deltacloud
- cloudservers-uk
- **vcloud (generic)**
- ec2 (generic)
- byon
- nova
- trmk-ecloud
- trmk-vcloudexpress

- **eucalyptus (generic)**
- cloudsigma-zrh
- elasticstack(generic)
- bluelock-vclouddirector
- slicehost
- **eucalyptus-partnercloud-ec2**
- elastichosts-lon-p (Peer 1)
- elastichosts-sat-p (Peer 1)
- elastichosts-lon-b (BlueSquare)
- openhosting-east1
- serverlove-z1-man
- skalicloud-sdg-my

BlobStore Example

```
BlobStoreContext context = new BlobStoreContextFactory().  
createContext(provider, identity, credential);
```

```
BlobStore blobStore = context.getBlobStore();  
blobStore.createContainerInLocation(null, containerName);
```

```
Blob blob = blobStore.newBlob("test");  
blob.setPayload("testdata");  
blobStore.putBlob(containerName, blob);
```

Amazon S3

AWS Management Console > Amazon S3 ALISSON W A SILVA | Help

Buckets

- Create Bucket
- Actions
- br.teste

Objects and Folders

br.teste

Upload | Create Folder | Actions

Refresh | Properties | Transfers | Help

Name	Size	Last Modified
The bucket 'br.teste' is empty		

Running BlobStore Example

Input:

```
java -jar target/blobstore-basics-jar-with-dependencies.jar aws-  
s3 <accesskey> <secretkey> <myfavoritecontainer>
```

Output:

```
14/10/2011 09:31:36 org.jclouds.logging.jdk.JDKLogger
```

```
logWarn
```

```
WARNING: hostname was br.teste.s3.amazonaws.com while  
session was br.teste.s3.amazonaws.com
```

```
br.teste: 1 entries
```

Amazon S3

AWS Management Console › Amazon S3 ALISSON W A SILVA ▾ | Help ▾

AWS
Elastic Beanstalk **S3** EC2 VPC CloudWatch Elastic MapReduce CloudFront CloudFormation RDS ElastiCache SQS IAM SNS

Buckets
Create Bucket Actions ▾
br.teste

Objects and Folders
Upload Create Folder Actions ▾ Refresh Properties Transfers Help

br.teste

Name	Size	Last Modified
test	8 bytes	Fri Oct 14 12:31:40 GMT-300 2011

ComputeService Example

```
Credentials login = new Credentials(System.getProperty("user.name"), Files.toString(new File(System.getProperty("user.home") + "/.ssh/id_rsa"), UTF_8));
```

```
ComputeService compute = new  
ComputeServiceContextFactory().createContext(provider,  
identity, credential, modules, properties).getComputeService();
```

```
// Default template chooses the smallest size on an operating  
system that tested to work with java, which tends to be Ubuntu  
or CentOS
```

```
TemplateBuilder templateBuilder = compute.templateBuilder();
```

ComputeService Example

// note this will create a user with the same name as you on the node. ex. you can connect via ssh publicip

```
Statement bootInstructions = AdminAccess.standard();
```

// to run commands as root, we use the runScript option in the template

```
templateBuilder.options(runScript(bootInstructions));
```

```
NodeMetadata node = getOnlyElement(compute.
```

```
createNodesInGroup(groupName, 1, templateBuilder.build()));
```

ComputeService Example

```
// when you run commands, you can pass options to decide
// whether to run it as root, supply or own credentials vs from
// cache, and wrap in an init script vs directly invoke
Map<? extends NodeMetadata, ExecResponse> responses =
compute.runScriptOnNodesMatching(
    inGroup(groupName), // predicate used to select nodes
    exec(command), // what you actually intend to run
    overrideCredentialsWith(login) // use my local user & ssh key
    .runAsRoot(false) // don't attempt to run as root (sudo)
    .wrapInInitScript(false)); // run command directly
```

Amazon EC2

Navigation

Region:

US East (Virginia)

EC2 Dashboard

INSTANCES

Instances

Spot Requests

Reserved Instances

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

NETWORK & SECURITY

Security Groups

Elastic IPs

Placement Groups

Load Balancers

Key Pairs

My Instances

Launch Instance Instance Actions

Show/Hide Refresh Help

Viewing: All Instances All Instance Types

1 to 10 of 10 Instances

	Name	Instance	AMI ID	Root Device	Type	Status	Security Groups	Key Pair Name	Monitoring	Virtualizati
<input type="checkbox"/>	br.teste	i-0848fe68	ami-8c1fece5	ebs	t1.micro	running	mygroup	ec2keypair	basic	paravirtual
<input type="checkbox"/>	empty	i-30d3e150	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#49	basic	paravirtual
<input type="checkbox"/>	empty	i-78dae818	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#98	basic	paravirtual
<input type="checkbox"/>	empty	i-f8d9eb98	ami-7f418316	ebs	t1.micro	terminated	mygroup	ec2keypair	basic	paravirtual
<input type="checkbox"/>	empty	i-c8deeca8	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#47	basic	paravirtual
<input type="checkbox"/>	empty	i-4a734d2a	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#38	basic	paravirtual
<input type="checkbox"/>	empty	i-4c68562c	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#76	basic	paravirtual
<input type="checkbox"/>	empty	i-e8675988	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#41	basic	paravirtual
<input type="checkbox"/>	empty	i-dc635dbc	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#66	basic	paravirtual
<input type="checkbox"/>	empty	i-285e6048	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#38	basic	paravirtual

0 EC2 Instances selected

Select an instance above

Running ComputeService Example

Input:

```
java -jar target/compute-basics-jar-with-dependencies.jar aws-ec2 <accesskey> <secretkey> mygroup add
```

ComputeService Example Output

```
Terminal
Arquivo  Editar  Ver  Terminal  Abas  Ajuda

Terminal
- >> running 1 node group(mygroup) location(us-east-1) image(us-east-1/ami-7341831a) hardwareProfile(t1.micro) options([groupIds=[], keyPair=null, noKeyPair=false, monitoringEnabled=false, placementGroup=null, noPlacementGroup=false, subnetId=null, userData=null, blockDeviceMappings=[], spotPrice=null, spotOptions=[formParameters={}]])
- >> searching params([biggest=false, fastest=false, imageName=null, imageDescription=Amazon Linux AMI x86 64 EBS, imageId=null, imagePredicate=null, imageVersion=2011.09.1, location=[id=us-east-1, scope=REGION, description=us-east-1, parent=aws-ec2, iso3166Codes=[US-VA], metadata={}], minCores=1.0, minRam=630, osFamily=amzn-linux, osName=null, osDescription=amazon/amzn-ami-2011.09.1.x86_64-ebs, osVersion=2011.09.1, osArch=paravirtual, os64Bit=true, hardwareId=null])
- << matched hardware(t1.micro)
- << matched image(us-east-1/ami-7341831a)
- >> creating keyPair region(us-east-1) group(mygroup)
- << created keyPair(jclouds#mygroup#us-east-1#59)
- >> creating securityGroup region(us-east-1) name(jclouds#mygroup#us-east-1)
- << created securityGroup(jclouds#mygroup#us-east-1)
- >> authorizing securityGroup region(us-east-1) name(jclouds#mygroup#us-east-1) port(22)
- << authorized securityGroup(jclouds#mygroup#us-east-1)
- >> authorizing securityGroup region(us-east-1) name(jclouds#mygroup#us-east-1) permission to itself
- << authorized securityGroup(jclouds#mygroup#us-east-1)
- >> running 1 instance region(us-east-1) zone(null) ami(ami-7341831a) params({InstanceType=[t1.micro], SecurityGroup.1=[jclouds#mygroup#us-east-1#59]})
- << started instances([region=us-east-1, name=i-3e4a745e])
- << present instances([region=us-east-1, name=i-3e4a745e])
- >> blocking on socket [address=184.72.164.72, port=22] for 600000 seconds
- << socket [address=184.72.164.72, port=22] opened
- Disabling high-strength ciphers: cipher strengths apparently limited by JCE policy
- >> running [sudo ./bootstrap init] as ec2-user@184.72.164.72
- << init(0)
- >> running [sudo ./bootstrap start] as ec2-user@184.72.164.72
- << start(1)
- << complete(true)
- << stdout from bootstrap as ec2-user@184.72.164.72
Reloading sshd: [ OK ]

- << stderr from bootstrap as ec2-user@184.72.164.72
useradd: warning: the home directory already exists.
Not copying any file from skel directory into it.

- << options applied node(us-east-1/i-3e4a745e)
<< node us-east-1/i-3e4a745e: [10.126.50.51, 184.72.164.72]
>> running [echo hello] on group mygroup as 05374909495
- >> blocking on socket [address=184.72.164.72, port=22] for 600000 seconds
- << socket [address=184.72.164.72, port=22] opened
- Disabling high-strength ciphers: cipher strengths apparently limited by JCE policy
- >> running [echo hello
] as 05374909495@184.72.164.72
- << [statements=[echo hello{lf}]](0)
<< node us-east-1/i-3e4a745e: [10.126.50.51, 184.72.164.72]
<< [output=hello
, error=, exitCode=0]
```

Amazon EC2

- Navigation
- Region: US East (Virginia)
- EC2 Dashboard
 - INSTANCES
 - Instances
 - Spot Requests
 - Reserved Instances
 - IMAGES
 - AMIs
 - Bundle Tasks
 - ELASTIC BLOCK STORE
 - Volumes
 - Snapshots
 - NETWORK & SECURITY
 - Security Groups
 - Elastic IPs
 - Placement Groups
 - Load Balancers
 - Key Pairs

My Instances

Launch Instance Instance Actions Show/Hide Refresh Help

Viewing: All Instances All Instance Types 1 to 11 of 11 Instances

	Name	Instance	AMI ID	Root Device	Type	Status	Security Groups	Key Pair Name	Monitoring	Virtualizati
<input type="checkbox"/>	br.teste	i-0848fe68	ami-8c1fece5	ebs	t1.micro	running	mygroup	ec2keypair	basic	paravirtual
<input type="checkbox"/>	empty	i-30d3e150	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#49	basic	paravirtual
<input type="checkbox"/>	empty	i-78dae818	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#98	basic	paravirtual
<input type="checkbox"/>	empty	i-f8d9eb98	ami-7f418316	ebs	t1.micro	terminated	mygroup	ec2keypair	basic	paravirtual
<input type="checkbox"/>	empty	i-c8deeca8	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#47	basic	paravirtual
<input type="checkbox"/>	empty	i-4a734d2a	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#38	basic	paravirtual
<input type="checkbox"/>	empty	i-4c68562c	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#76	basic	paravirtual
<input type="checkbox"/>	empty	i-e8675988	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#41	basic	paravirtual
<input type="checkbox"/>	empty	i-dc635dbc	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#66	basic	paravirtual
<input type="checkbox"/>	empty	i-285e6048	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#38	basic	paravirtual
<input type="checkbox"/>	empty	i-3e4a745e	ami-7341831a	ebs	t1.micro	running	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#59	basic	paravirtual

0 EC2 Instances selected

Select an instance above

Running ComputeService Example

Input:

```
java -jar target/compute-basics-jar-with-dependencies.jar aws-ec2 <accesskey> <secretkey> mygroup exec "echo hello"
```

Running ComputeService Example

Output:

```
>> initialized provider [id=aws-ec2, endpoint=https://ec2.us-east-1.
amazonaws.com, apiVersion=2010-06-15,
identity=AKIAJASGCLQOJJDTOTAA, iso3166Codes=[US-VA, US-CA, IE, SG,
JP-13]]
>> running [echo hello] on group mygroup as 05374909495
- >> blocking on socket [address=107.20.5.232, port=22] for 600000 seconds
- << socket [address=107.20.5.232, port=22] opened
- Disabling high-strength ciphers: cipher strengths apparently limited by JCE
policy
- >> running [echo hello
] as 05374909495@107.20.5.232
- << [statements=[echo hello{lf}]](0)
<< node us east 1/i 1c310f7c: [10.112.49.244, 107.20.5.232]
<< [output=hello
, error=, exitCode=0]
```

Running ComputeService Example

Input:

```
java -jar target/compute-basics-jar-with-dependencies.jar aws-ec2 <accesskey> <secretkey> mygroup destroy
```


Amazon EC2

- Navigation
- Region: US East (Virginia)
- EC2 Dashboard
 - INSTANCES
 - Instances
 - Spot Requests
 - Reserved Instances
 - IMAGES
 - AMIs
 - Bundle Tasks
 - ELASTIC BLOCK STORE
 - Volumes
 - Snapshots
 - NETWORK & SECURITY
 - Security Groups
 - Elastic IPs
 - Placement Groups
 - Load Balancers
 - Key Pairs

My Instances

Launch Instance Instance Actions Show/Hide Refresh Help

Viewing: All Instances All Instance Types 1 to 11 of 11 Instances

	Name	Instance	AMI ID	Root Device	Type	Status	Security Groups	Key Pair Name	Monitoring	Virtualization
<input type="checkbox"/>	br.teste	i-0848fe68	ami-8c1fece5	ebs	t1.micro	running	mygroup	ec2keypair	basic	paravirtual
<input type="checkbox"/>	empty	i-30d3e150	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#49	basic	paravirtual
<input type="checkbox"/>	empty	i-78dae818	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#98	basic	paravirtual
<input type="checkbox"/>	empty	i-f8d9eb98	ami-7f418316	ebs	t1.micro	terminated	mygroup	ec2keypair	basic	paravirtual
<input type="checkbox"/>	empty	i-c8deeca8	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#47	basic	paravirtual
<input type="checkbox"/>	empty	i-4a734d2a	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#38	basic	paravirtual
<input type="checkbox"/>	empty	i-4c68562c	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#76	basic	paravirtual
<input type="checkbox"/>	empty	i-e8675988	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#41	basic	paravirtual
<input type="checkbox"/>	empty	i-dc635dbc	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#66	basic	paravirtual
<input type="checkbox"/>	empty	i-285e6048	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#38	basic	paravirtual
<input type="checkbox"/>	empty	i-3e4a745e	ami-7341831a	ebs	t1.micro	terminated	jclouds#mygroup#us-east-1	jclouds#mygroup#us-east-1#59	basic	paravirtual

0 EC2 Instances selected

Select an instance above

More information

Installation Guide:

<http://code.google.com/p/jclouds/wiki/Installation>

Documentation:

<http://www.jclouds.org/documentation/index>

<http://code.google.com/p/jclouds/>

Javadocs:

<http://www.jclouds.org/jclouds-maven-site/1.1.1/jclouds-multi/apidocs/>

<http://jclouds.rimuhosting.com/apidocs/>

More information

Users Group:

<https://groups.google.com/forum/#!forum/jclouds>

Developers Group:

<https://groups.google.com/forum/#!forum/jclouds-dev>

Twitter:

<http://twitter.com/#!/jclouds>

Contributing to jclouds

Develop and extend jclouds code:

<http://www.jclouds.org/documentation/devguides/contributing-to-jclouds>

Develop and extend jclouds documentation:

<http://www.jclouds.org/documentation/devguides/contributing-to-documentation>

Obrigado!

<http://jclouds.org/>

palestrante: Alisson Wilker

e-mail: alisson-wilker.silva@serpro.gov.br

19 de outubro de 2011