



API Analyse

Silvan Adrian
Fabian Binna

1 Änderungshistorie

Datum	Version	Änderung	Autor
25.09.15	1.00	Erstellung des Dokuments	Gruppe
25.09.15	1.01	APIs	Fabian Binna
25.09.15	1.02	Support	Fabian Binna
25.09.15	1.03	Fazit	Fabian Binna
25.09.15	1.04	Einführung + Gültigkeit	Fabian Binna

Inhaltsverzeichnis

1	Änderungshistorie	2
2	Einführung	4
2.1	Zweck	4
2.2	Gültigkeitsbereich	4
2.3	Referenzen	4
3	APIs	5
3.1	Libcloud	5
3.2	jClouds	5
3.3	elibcloud	5
3.4	fog	5
3.5	pkgcloud	5
4	Support	6
4.1	Compute	6
4.2	Storage (Object/Blob)	6
4.3	Network	7
4.4	Other	7
4.4.1	Database	7
4.4.2	DNS	7
4.4.3	Load Balancer	8
4.4.4	Orchestration	9
4.4.5	CDN	9
5	Fazit	10
5.1	libcloud	10
5.2	jclouds	10
5.3	fog	10
5.4	pkgcloud	11

2 Einführung

2.1 Zweck

Dieses Dokument beinhaltet die Analyse von APIs die den Zugriff auf diverse Cloud Anbieter ermöglicht.

2.2 Gültigkeitsbereich

Dieses Dokument ist während des ganzen Projekts gültig.

2.3 Referenzen

Libcloud
jClouds
elasticsearch
fog
pkgcloud

3 APIs

3.1 Libcloud

Sprache: Python

Wichtigste Provider: Rackspace, Amazon web services, CloudStack, OpenStack, DigitalOcean, Eucalyptus, Joyent, Linode, exoscale, NephoScale, Google Cloud Platform, Zerigo, CloudSigma, iKoula, libvirt

3.2 jClouds

Sprache: Java

Wichtigste Provider: OpenStack, Docker, DigitalOcean, Google Cloud Platform, Rackspace, HP Cloud, CloudStack, Amazon web services, abiquo, CloudSigma, joyent

3.3 elibcloud

Sprache: Erlang

elibcloud ist ein Wrapper für libcloud.

3.4 fog

Sprache: Ruby

Wichtigste Provider: CloudSigma, CloudStack, GoGrid, Google Cloud Platform, Joyent, Libvirt, Linode, OpenStack, OpenVZ, Rackspace, Zerigo, IBM, HP

3.5 pkgcloud

Sprache: JavaScript (Node.js)

Wichtigste Provider: Amazon, Azure, DigitalOcean, Joyent, OpenStack, Rackspace, Google, HP,

4 Support

4.1 Compute

Die grösste Auswahl an Providern liefert Libcloud.JClouds hingegen unterstützt auch Docker, was ein grosser Vorteil gegenüber Libcloud ist. Im Dokument Compute.ods im Ordner 02_Analyse/01_API wird genau aufgeführt, welche Provider von welchen APIs unterstützt werden. Es werden nur public Clouds berücksichtigt.

4.2 Storage (Object/Blob)

libcloud

- PCextreme AuroraObjects
- Microsoft Azure (blobs)
- CloudFiles
- Google Storage
- KTUCloud Storage
- Numbus.io
- Ninefold
- OpenStack Swift
- Amazon

jclouds (BlobStore)

- AWS
- HP Helion
- Azure
- Rackspace

fog

- S3
- CloudFiles
- Google Storage

pkgcloud

- Amazon

- Azure
- Google
- HP
- OpenStack
- Rackspace

4.3 Network

Alle APIs ausser pkgcloud erwähnen keine Provider für die Unterstützung von Network Providern. Bei libcloud sind jedoch Methoden vorhanden, die auf eine Netzwerkkonfigurationsmöglichkeit hinweisen, werden aber nicht genauer dokumentiert.

pkgcloud

- HP
- OpenStack
- Rackspace

4.4 Other

4.4.1 Database

pkgcloud

- IrisCouch
- MongoLab
- Rackspace
- MongoHQ
- RedisToGo

4.4.2 DNS

libcloud

- AuroraDNS
- DigitalOcean
- Gandi
- Google

- Host Virtual
- Linode
- Rackspace
- AWS Route53
- Softlayer
- Zerigo

fog

- AWS Route53
- Blue Box
- DNSimple
- Linode
- Rackspace
- Rage4
- Slicehost
- Zerigo

pkgcloud

- Rackspace

4.4.3 Load Balancer

libcloud

- Brightbox
- CloudStack
- DimensionData
- Amazon
- Google
- GoGrid
- Ninefold
- Rackspace

- Softlayer

jclouds

- AWS Elastic LoadBalancer
- Rackspace

pkgcloud

- Rackspace

4.4.4 Orchestration

pkgcloud (beta)

- OpenStack
- Rackspace

4.4.5 CDN

fog

- CloudFront

5 Fazit

Im Gesamtbild schneidet libcloud am besten ab. Es bietet deutlich am meisten Compute und Storage Provider. Die Dokumentation ist sehr ausführlich, mit konkreten Ratschlägen zur Implementation (z.B. Thread Safe). Zusätzlich bietet libcloud Module für SSL und Pricing. Jclouds ist eine Library für Java, was für uns am besten ist, da wir am meisten Erfahrung mit Java haben. Es gibt jedoch nicht viele Compute Provider, dafür unterstützt jclouds als einziger Docker. Der einzige Vorteil von fog ist die Möglichkeit CDNs als Service anzubieten. Pkgcloud unterstützt eine breite Auswahl von Services (z.B. Database, Load Balancer, DNS). Elibcloud ist ein erlang Wrapper für libcloud und unterstützt somit das gleiche wie libcloud (sonlange die Version auf dem neusten stand ist). Erlang würde sich für eine parallele Umgebung eignen, es sind jedoch keinerlei Erlang Kenntnisse im Team vorhanden.

Wir entscheiden uns für libcloud. Die Gründe dafür sind unten aufgeführt

5.1 libcloud

- ⊕ Grösste Auswahl an Compute und Storage Provider.
- ⊕ Am besten dokumentiert. Für jede Methode existiert eine Tabelle, die zeigt welche Provider damit angesprochen werden können.
- ⊕ Ist zwar nicht Thread-Safe. Es werden jedoch konkrete Lösungsvorschläge gemacht.
- ⊕ SSL und Pricing Module vorhanden.
- ⊖ Team hat wenig Erfahrung mit komplexen/grossen Python Projekten.

5.2 jclouds

- ⊕ Unterstützt Docker.
- ⊕ Java Librariy. Das Team hat am meisten Erfahrung mit Java.
- ⊕ Code Examples für fast jeden Provider.
- ⊖ Kleine Auswahl an Compute Providern.

5.3 fog

- ⊕ Es ist möglich ein CDN als Service anzubieten.
- ⊖ Mässige Dokumentation. Es existieren zwar Examples, die sind aber nicht besonders aussagekräftig.
- ⊖ Kleine Auswahl an Compute Providern.

5.4 pkgcloud

- ⊕ Grösste Auswahl an Services.
- ⊕ Database as a Service
- ⊕ Orchestration
- ⊕ Explizite Unterstützung von Network.
- ⊖ Mässige Dokumentation. Es existieren zwar Examples, die sind aber nicht besonders aussagekräftig.
- ⊖ Kleine Auswahl an Compute Providern.