

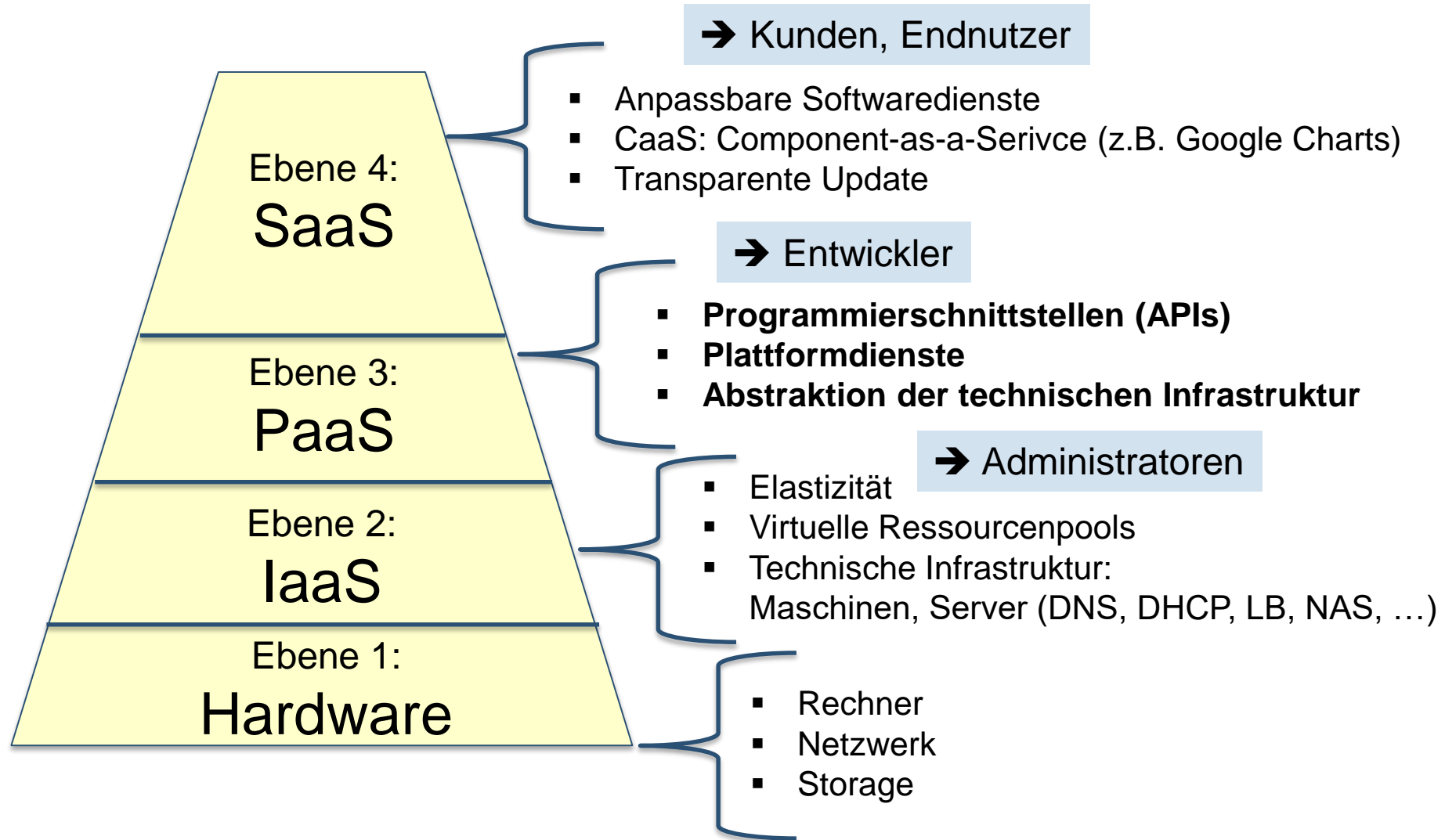
Cloud Computing

Kapitel 10: Platform-as-a-Service

Dr. Josef Adersberger

Grundlagen zu einer PaaS Cloud

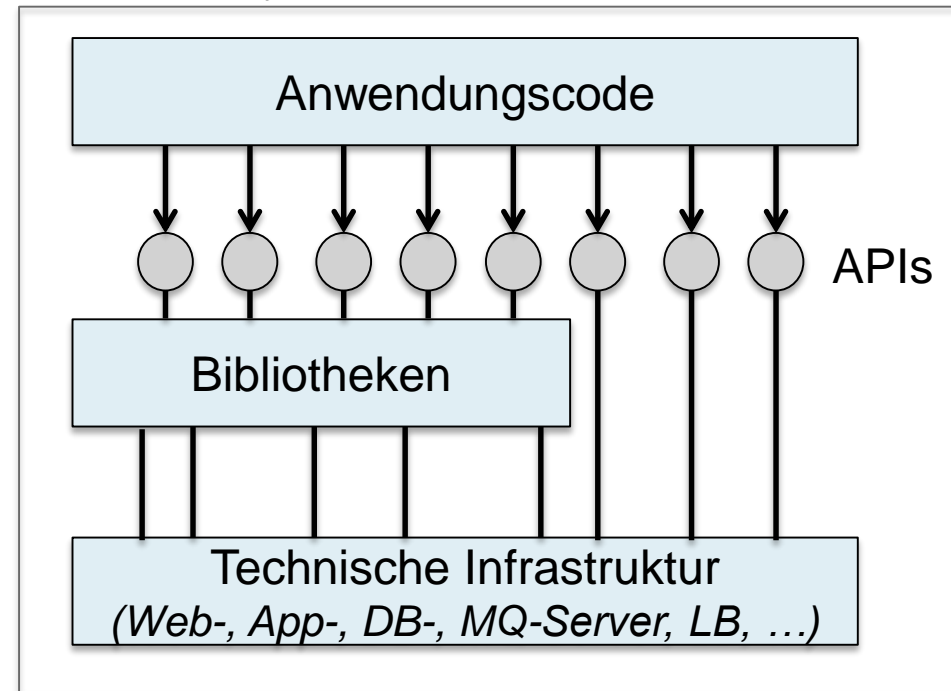
Das Schichtenmodell des Cloud Computing: Vom Blech zur Anwendung.



Das Problem: Stovepipe Architecture. Anwendungen aufwändig von Hand verdrahten.

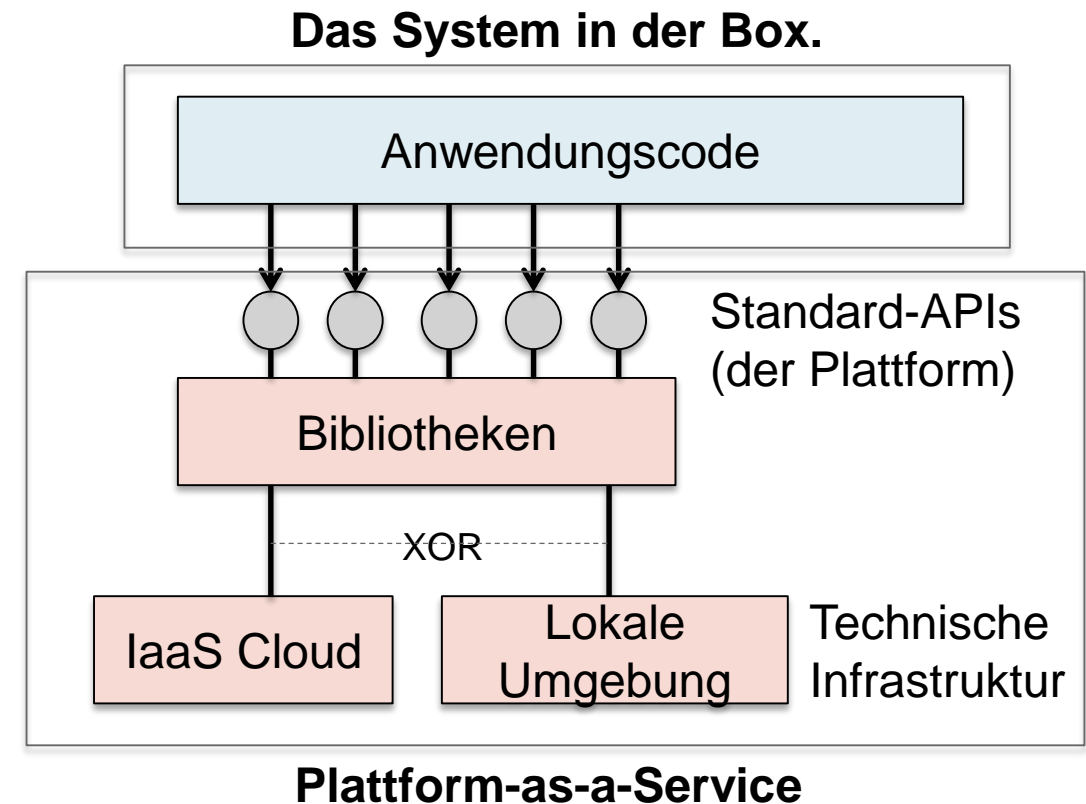


Das System: Mühevoll verdrahtet.



Die Lösung: Plattform-as-a-Service bietet eine ad-hoc Entwicklungs- und Betriebsplattform.

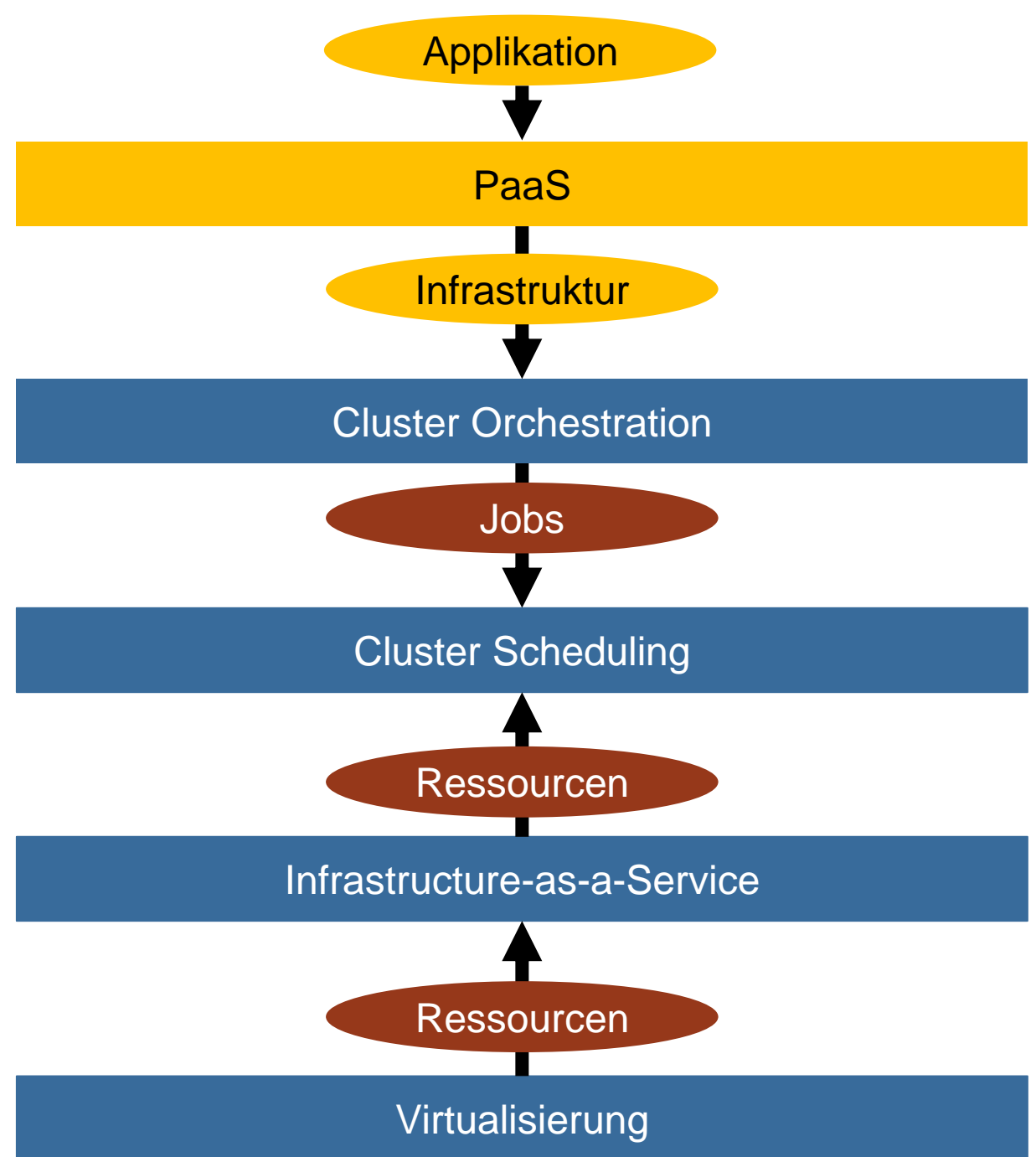
- Die Anwendung sieht nur Programmierschnittstellen seiner Laufzeitumgebung.
„Engine and Operating System should not matter....”.
- Die Anwendung wird per Applikationspaket deployed. Es ist kein Image mit Technischer Infrastruktur notwendig.
- Es erfolgt eine automatische Skalierung der Anwendung.
- PaaS repräsentiert einen Vertrag zwischen Entwicklung und Betrieb.
- Die Programmierschnittstellen (APIs) stehen als Bibliotheken oder als (REST-) Webservices zur Verfügung.
- Entwicklungswerkzeuge (insb. Plugins für IDEs und Buildsysteme sowie eine lokale Testumgebung) stehen zur Verfügung: „deploy to cloud“.
- Die Plattform bietet eine Schnittstelle zur Administration und zum Monitoring der Anwendungen.



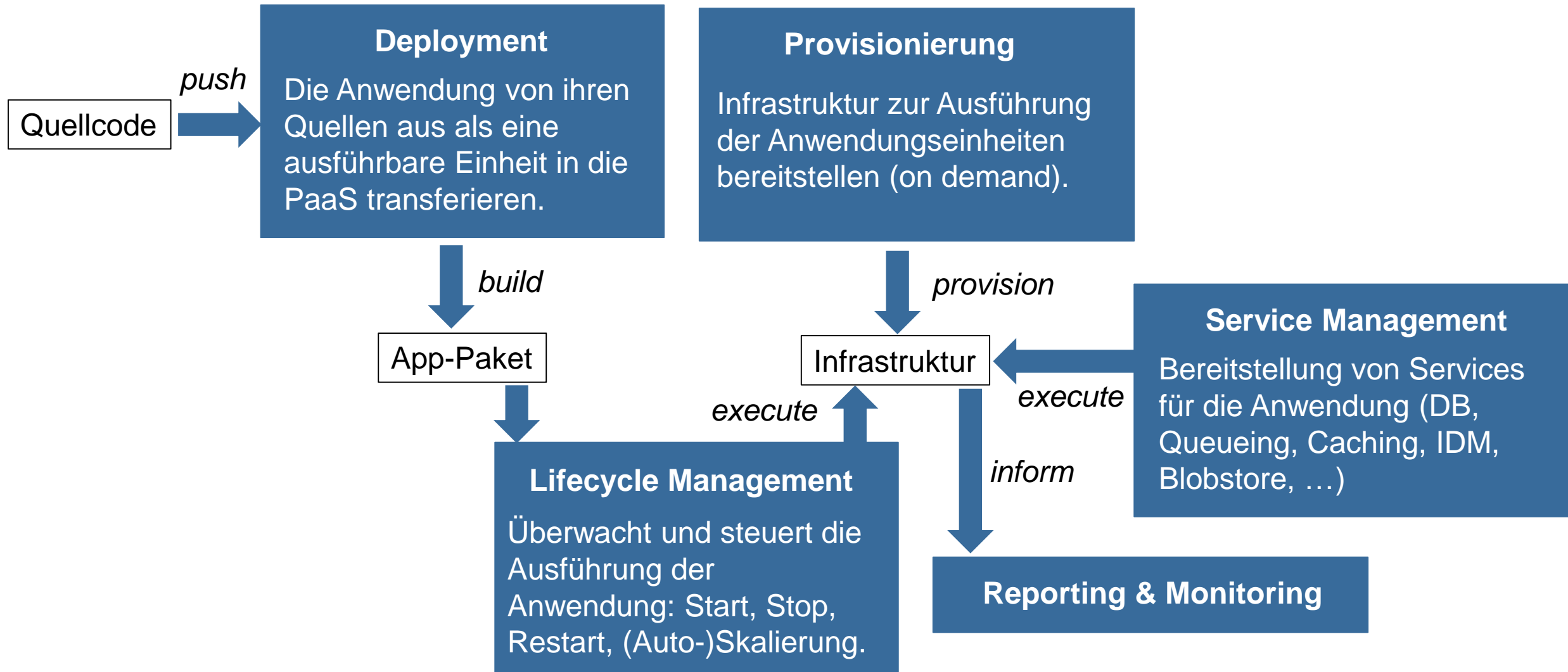
PaaS: Definitionen

- NIST: The capability provided to the consumer is to **deploy onto the cloud infrastructure** consumer-created or acquired applications created **using programming languages, libraries, services, and tools supported by the provider**. **The consumer does not manage or control the underlying cloud infrastructure** including network, servers, operating systems, or storage, **but has control over the deployed applications** and possibly configuration settings for the application-hosting environment.
- Forrester: A complete **application platform** for multitenant cloud environments that **includes development tools, runtime, and administration** and management tools and services. PaaS combines an **application platform with managed cloud infrastructure** services.

Das Big Picture

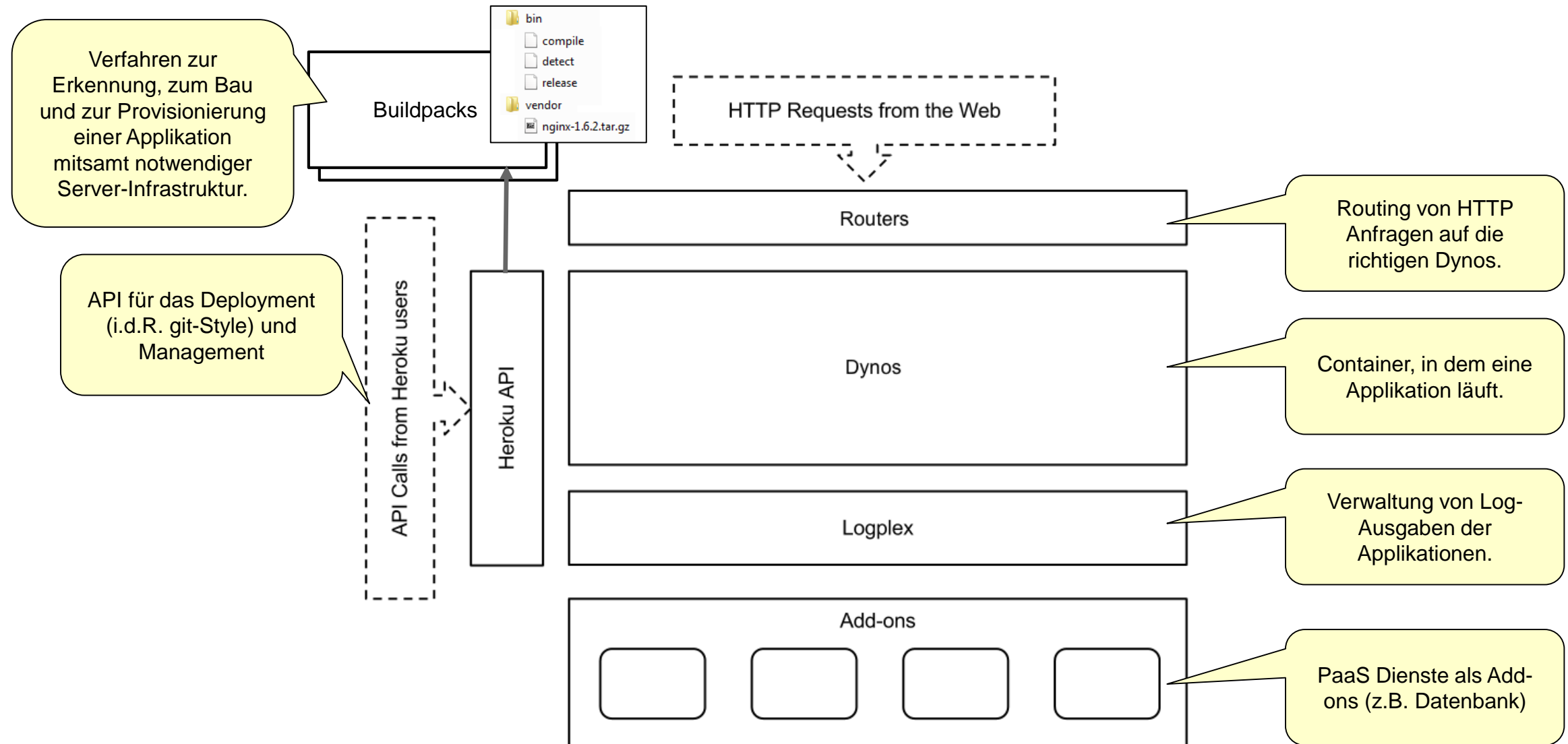


Die funktionalen Building Blocks einer PaaS Cloud.

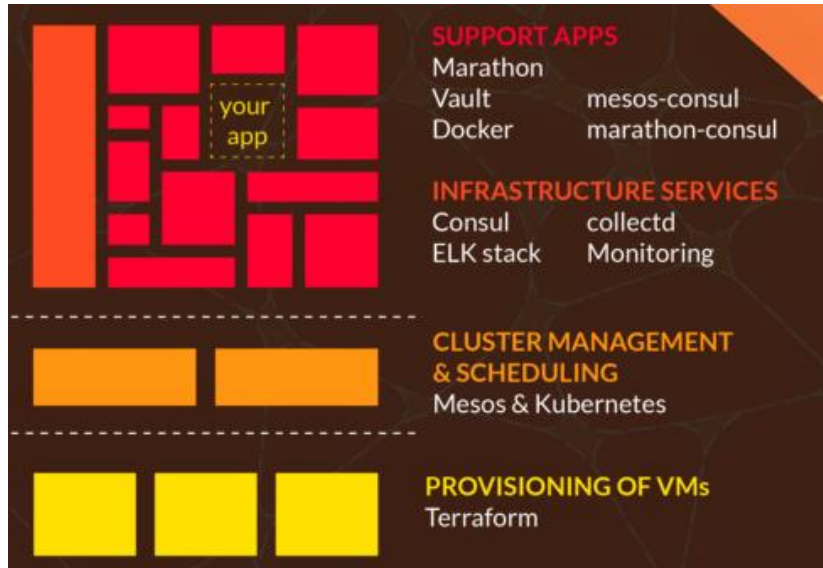


← = Datenfluss

High-Level Architektur einer PaaS am Beispiel Heroku.



Die technischen Building-Blocks von PaaS-Lösungen: Sehen sie die Gemeinsamkeiten?



Quelle: <https://mantl.io>

Quelle: <https://github.com/yelp/paasta>

Note: PaaS is an opinionated platform that uses a few un-opinionated tools. It requires a non-trivial amount of infrastructure to be in place before it works completely:

- [Docker](#) for code delivery and containment
- [Mesos](#) for code execution and scheduling (runs Docker containers)
- [Marathon](#) for managing long-running services
- [Chronos](#) for running things on a timer (nightly batches)
- [SmartStack](#) for service registration and discovery
- [Sensu](#) for monitoring/alerting
- [Jenkins](#) (optionally) for continuous deployment



Apollo is built on top of the following components:

- [Packer](#) for automating the build of the base images
- [Terraform](#) for provisioning the infrastructure
- [Apache Mesos](#) for cluster management, scheduling and resource isolation
- [Consul](#) for service discovery, DNS
- [Docker](#) for application container runtimes
- [Weave](#) for networking of docker containers
- [HAProxy](#) for application container load balancing

Quelle: <https://github.com/Capgemini/Apollo>

Cloud-fähige Softwarearchitektur

Cluster Orchestration

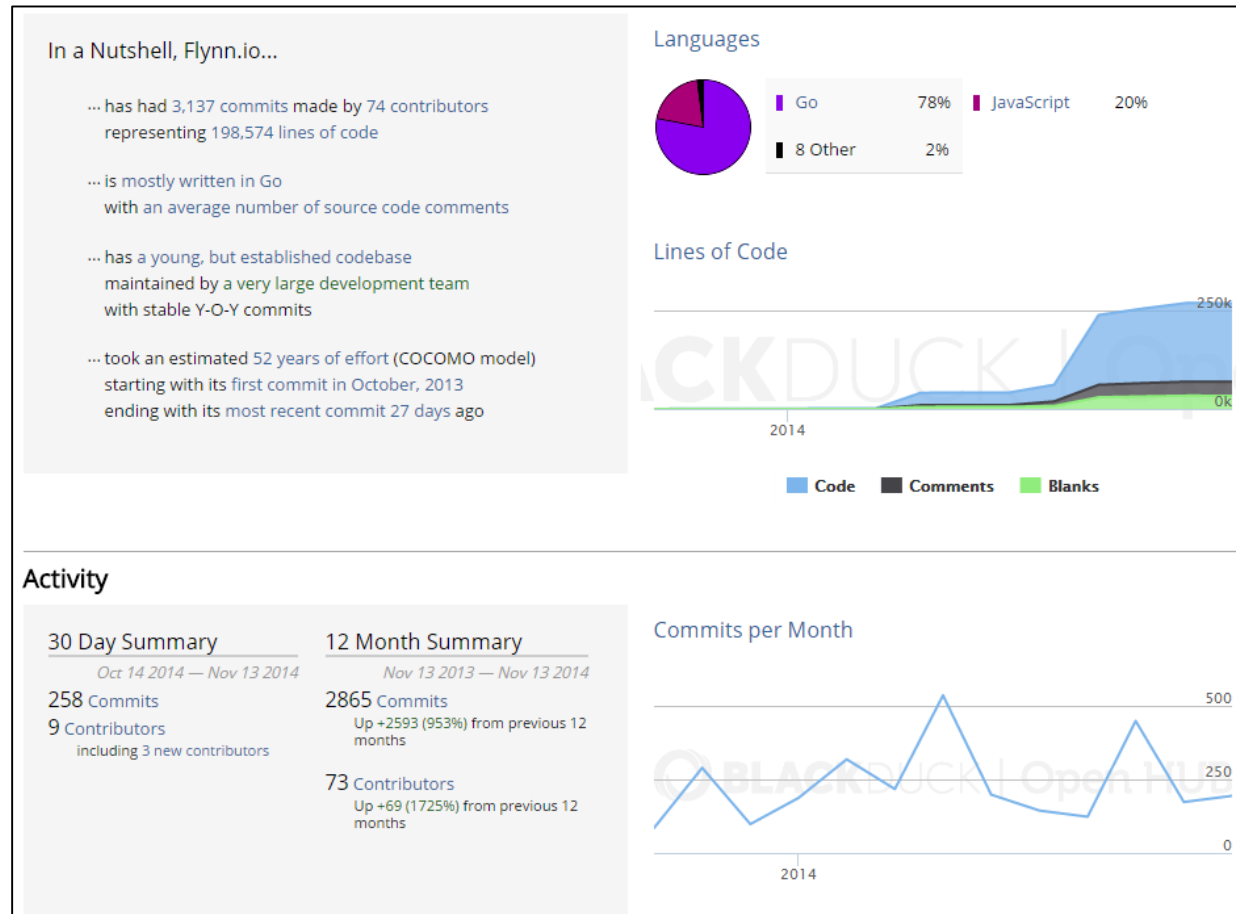
Cluster Scheduling

Private PaaS Clouds am Beispiel Flynn

Flynn

■ Private PaaS auf Basis Docker

■ Open-Source-Projekt unter einer BSD Lizenz



The product that ops provides to developers

Ops should be a product team, not consultants. Flynn is the single platform that ops can provide to developers to power production, testing, and development, freeing developers to focus.

[Learn More](#)[Try now](#)

```
remote: Installing Bazaar... done
remote: -----> Running: go get -tags heroku ./...
remote: -----&gt; Discovering process types
remote: Profile declares types -> web
remote: -----> Build complete!
remote: -----> Deploying minos ...
remote: -----> Application deployed:
remote: http://minos.mixie.is
remote: ----->
To git@mixie.is:minos
* [new branch] master -> master
minos git:(master) curl -s minos.mixie.is |cowthink
( This is pretty fantastic! )
  o  ^__^
  o  (oo)\_____
    (_____.       )\/\
      ||----w |
      ||     ||
minos git:(master)
```

Flynn is a set of components for solving ops.

Flynn components are divided into two layers.

Layer 0 is a low-level resource framework including service discovery inspired by the [Google Omega](#) paper.

Layer 1 is a set of higher level components that makes it easy to deploy, maintain, and scale applications and databases.

[Learn more about Flynn's architecture](#)

Easy deployment

Flynn lets you deploy apps with git push and Docker containers. Developers can deploy any app to any cluster in seconds.

Run anything

Flynn goes beyond 12 factor apps. Run any Linux process written in any language or framework, even stateful apps on your own servers or any public cloud.

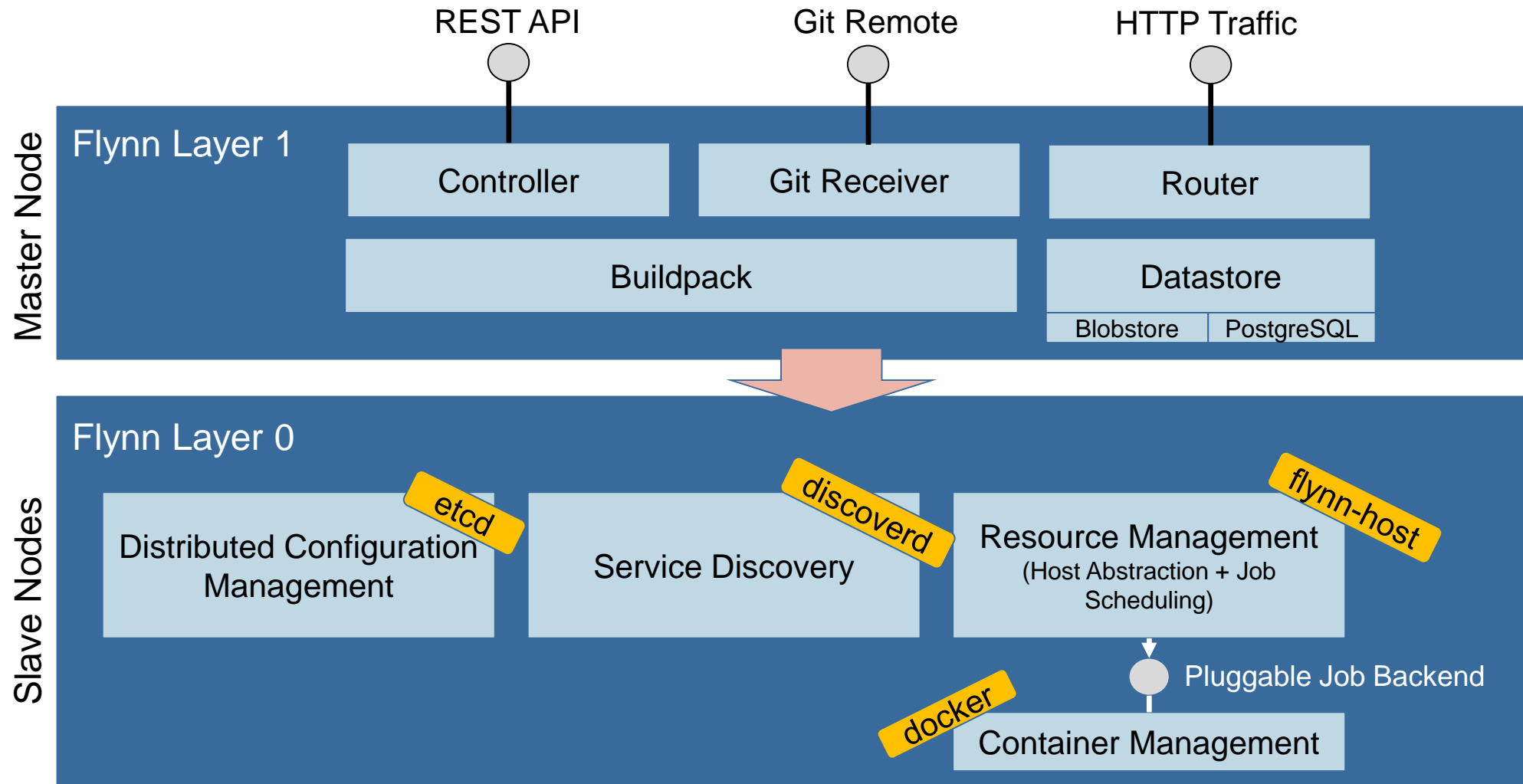
Painless scaling

Scaling or adding a new cluster is simple: just add more nodes. Everything is containerized; Flynn takes care of distributing work across the cluster.

Simple and composable

Flynn is 100% free and open source. Flynn works great out of the box, and since Flynn is modular and API-driven it's easy to modify and swap components to suit your needs.

Die Architektur von Flynn



Alternative Private PaaS Clouds

- Deis (<http://deis.io>, zu Flynn vergleichbarer Ansatz auf Basis von Docker)
- OpenShift (<https://www.openshift.com>, PaaS mit Schwerpunkt JEE von Red Hat)
- CloudFoundry (<http://www.cloudfoundry.org>, produktionserprobte PaaS von Pivotal mit breiter Unterstützung aus der Industrie)
- Stackato (<http://www.activestate.com/stackato>). Private PaaS von ActiveState (kommerziell).
- PaaSSTA (<https://github.com/yelp/paasta>). Open-Source private PaaS auf Basis von Mesos und Marathon.
- VAMP (<http://vamp.io>). Leichtgewichtige Open-Source private PaaS ausgelegt auf Microservices. Läuft auf Basis Mesos oder Kubernetes.

Public PaaS Clouds am Beispiel Google App Engine

Ein PaaS-Vergleich über die angebotenen APIs und Services.

	GAE-J	AWS
Datenspeicher	App Engine Datastore (Key/Value mit JDO und JPA API) Cloud Storage (Objekte) Blobstore (Dateien), Cloud SQL (relational)	DynamoDB (Key/Value), S3 (Objekte und Dateien), RDS (relational)
Messaging	Mail (mit javax.mail API), XMPP, Channel (Push-API)	SES (E-Mails), SNS (Notifications), SQS (Message Queuing)
Engine	Servlet Engine, Capabilities, LogService	Elastic Beanstalk (Servlet Engine)
Integration	URLFetch, App Identity, OAuth	
Parallele Verarbeitung	Task Queue	Elastic MapReduce
Volltextsuche	Search, Prospective Search	CloudSearch
Cache	Memcache mit JCache-API	ElastiCache
User-Authentifizierung	Google Accounts, OpenID	IAM
SaaS-APIs	Google Data API, Images, Conversion	SWF (Workflows)
Mandantenfähigkeit	Multitenancy (Namespaces API)	

Die Google App Engine

- Die Google App Engine (GAE) ist das PaaS-Angebot von Google.
- Anwendungen laufen innerhalb der Google Infrastruktur.
- Der Betrieb der Anwendungen ist innerhalb bestimmter Quoten kostenfrei. Danach fallen Kosten u.A. auf Basis von Service-Aufrufen, Storage-Volumen und real genutzten CPU-Sekunden an.
- Unterstützte Sprachen:
 - Python (seit 2008)
 - Java (seit 2009)
 - Go - „Python meets C++“ (seit 2011)
 - PHP (seit 2013)
- Integrationen in alle gängigen IDEs stehen zur Verfügung (Eclipse, IntelliJ, Netbeans).



Ausgewählte GAE Services (1/2)

■ Datastore

- Persistenter Speicher, realisiert als Key/Value-Datenbank.
- Transaktionen sind atomar. Schreibvorgänge sind stark konsistent. Abfragen sind eventuell konsistent.
- Definition, Abfrage und Manipulation von Daten erfolgt über eine eigene Sprache, die GQL (Google Query Language, nah an SQL).
- Als High-Level API sind die JDO und JPA APIs verfügbar. Diese sind im Rahmen von Java/JEE standardisiert. Die API wird durch das DataNucleus-Framework implementiert.

■ Memcache

- Hochperformanter temporärer Datenspeicher im Hauptspeicher (In-Memory Data-Grid).
- Jeder Eintrag wird mit einem eindeutigen Schlüssel abgelegt.
- Jeder Eintrag ist auf 1 MB beschränkt.
- Es wird eine Verfallszeit in Sekunden angegeben, wann der Eintrag aus dem Memcache entfernt werden soll.
- Daten werden je nach Auslastung des Memcache auch bereits früher verdrängt.
- Als High-Level API ist die JCache API verfügbar.

Ausgewählte GAE Services (2/2)

■ URL Fetch

- Zugriff auf Inhalte im Internet.
- Unterstützte Methoden: GET, POST, PUT, DELETE und HEADER.
- Es darf auf Ports in den Bereichen 80-90, 440-450 und 1024-65535 zugegriffen werden.
- Anfragen und Antworten sind auf jeweils 1 MB beschränkt.
- Timeout für URL-Anfragen ist 5 Sekunden.

■ Users

- Anbindung eines Single-Sign-On Systems.
- Es werden Google Accounts und OpenID Accounts unterstützt.
- Als High-Level-API wird JAAS genutzt.

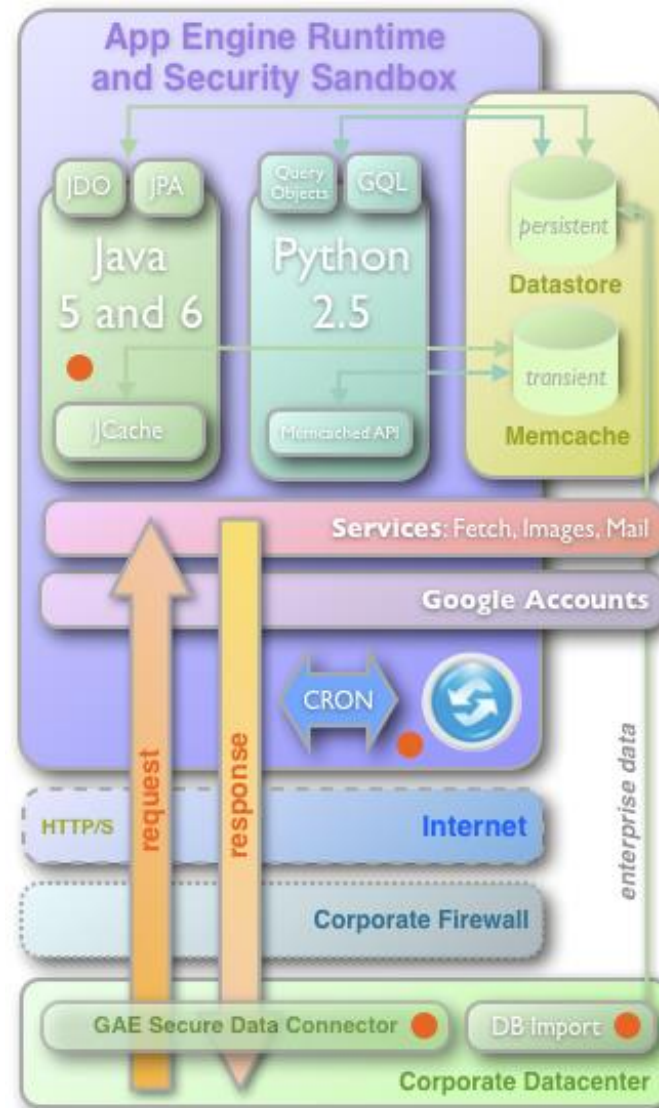
■ XMPP

- Nachrichten können an jedes XMPP-kompatibles Nachrichtensystem (z.B. Google Talk) gesendet und von diesem empfangen werden.
- Jede Anwendung besitzt einen eindeutigen XMPP-Benutzernamen.

Alle APIs:

- [App Identity](#)
- [Blobstore](#)
- [Google Cloud Storage](#)
- [Capabilities](#)
- [Channel](#)
- [Conversion](#)
- [Images](#)
- [Mail](#)
- [Memcache](#)
- [Multitenancy](#)
- [OAuth](#)
- [Prospective Search](#)
- [Search](#)
- [Task Queues](#)
- [URL Fetch](#)
- [Users](#)
- [XMPP](#)

Die Google App Engine im Überblick.




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

Einschränkungen der Google App Engine.

Eine GAE-Applikation läuft in einer Sandbox, die das Verhalten der Applikation einschränkt. Dies geschieht mit dem Ziel, die Verarbeitung effizient zu halten und die Infrastruktur im Auto-Scaling zu schützen.




- Es dürfen nicht alle Klassen der Standardbibliothek genutzt werden
 - Keine eigenen Threads öffnen
 - Kein Zugriff auf die Laufzeitumgebung und z.B. ihre Classloader
 - <http://code.google.com/p/googleappengine/wiki/WillItPlayInJava>
- Kommunikation mit anderen Web-Anwendungen oder Servern nur über URL Fetch, XMPP oder Email
 - Anfragen und Antworten dürfen maximal 1MB groß sein
 - Web-Hooks als allgemeines Architekturmittel für eingehende Kommunikation. Angestoßen bei Ereignissen (Warmup), Messages oder Cron-gesteuert.
- Alle Requests an eine GAE-Anwendung werden nach 60 Sekunden beendet
- Diverse Einschränkungen zu Datenvolumina und Anzahl von Service-Aufrufen

Funktionen der Admin-UI: Die Applikationsübersicht.

 adersberger@googlemail.com | [My Account](#) | [Help](#) | [Sign out](#)

My Applications

« Prev 20 1-3 of 3 Next 20 »

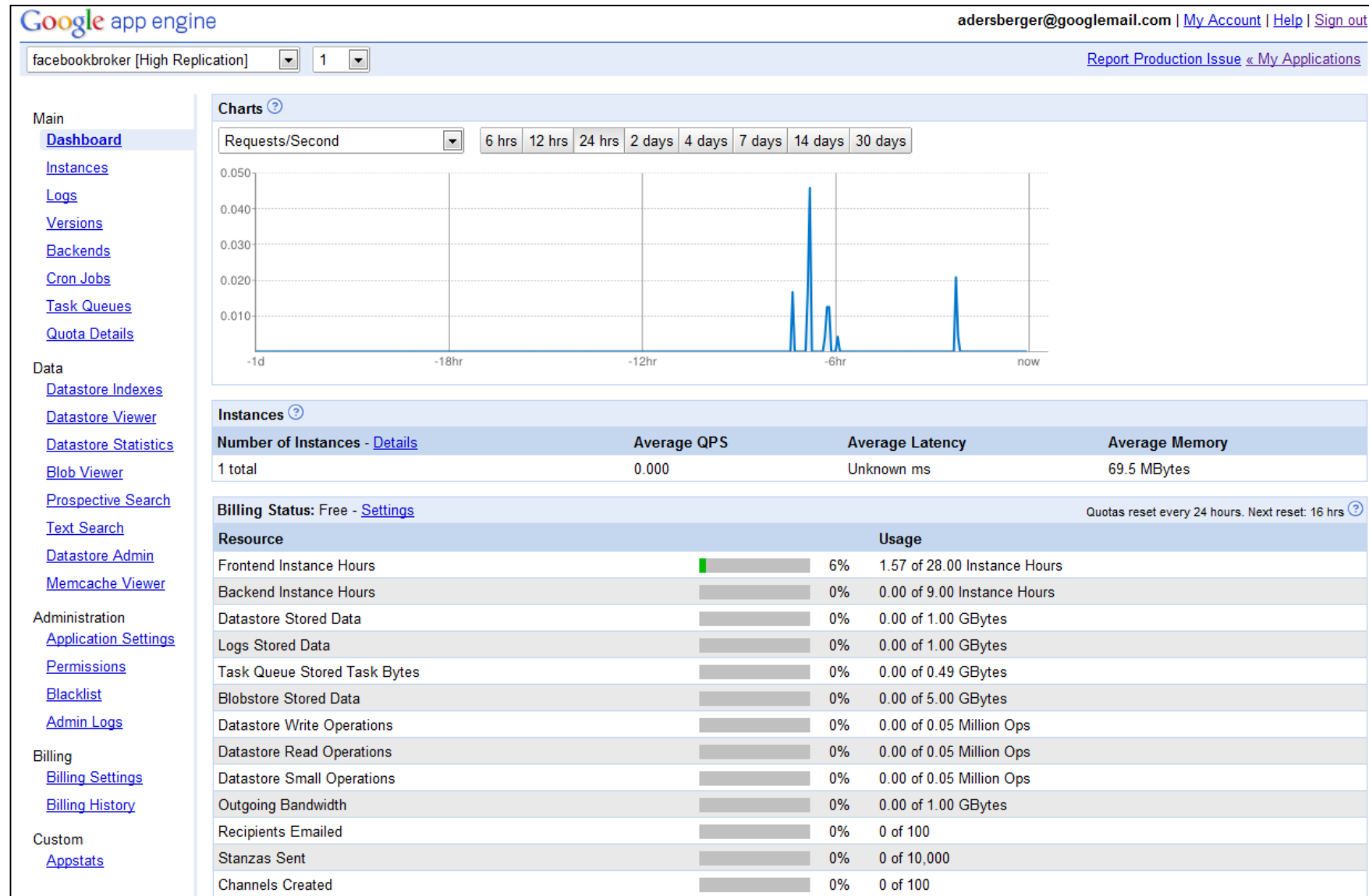
Application	Title	Billing Administrator	Storage Scheme	Current Version
cczwitscher	Zwitscher 4 Cloud Computing		High Replication	1 
facebookbroker	Facebook Broker		High Replication	2 
qaware-cc-sandbox	QAware Cloud Computing Sandbox		High Replication	1 

Create Application

« Prev 20 1-3 of 3 Next 20 »

You have 7 applications remaining.

Funktionen der Admin-UI: Das Applikations-Dashboard.



Funktionen der Admin-UI: Zugriff und Analyse der Log-Einträge.

Total Logs Storage: 47 KBytes spanning 1 day, 3 hours (0% of the Retention limit) **Total Logs Storage for Version:** 47 KBytes (99% of Logs Storage) [Change Settings](#)

Show: ☒ All requests ☐ Logs with minimum severity:

Error

Timezone:

(GMT-7:00) US/Pacific

[Options](#)

Tip: Click a log line to show or hide its details. [Expand logs](#)

[Prev 20](#) **21-40** [Next 20](#) (Top: 6:55:44 ago)

Last record searched: 06-17 05:27AM 53.952. Use Next link to search older records.

+	2012-06-18 01:30:41.354	/appstats/static/appstats_js.js	200	81ms	44kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
+	2012-06-18 01:30:41.351	/appstats/static/app_engine_logo_sm.gif	200	69ms	3kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
+	2012-06-18 01:30:41.348	/appstats/static/appstats_css.css	200	78ms	15kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
+	2012-06-18 01:30:41.118	/appstats/stats	200	523ms	7kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
+	2012-06-18 01:30:40.423	/appstats	302	37ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
+	2012-06-18 01:30:39.452	/appstats	302	25ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
+	2012-06-18 01:29:49.458	/resources/facebook/Java	200	5407ms	3kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
+	2012-06-18 01:29:43.453	/resources/facebook/Java	303	19915ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
I	2012-06-18 01:29:43.450	This request caused a new process to be started for your application, and thus caused your application code to be loaded for the first time. This requ				
+	2012-06-18 01:29:20.878	/	200	7ms	0kb	
+	2012-06-18 01:00:25.876	/resources/facebook/Java	200	18993ms	83kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
I	2012-06-18 01:00:25.866	This request caused a new process to be started for your application, and thus caused your application code to be loaded for the first time. This requ				
+	2012-06-18 01:00:05.132	/	304	7ms	0kb	
+	2012-06-18 01:00:01.921	/	304	10ms	0kb	
+	2012-06-18 00:59:55.130	/	200	116ms	0kb	
+	2012-06-17 06:19:25.565	/resources/facebook/Java	200	16954ms	2kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
I	2012-06-17 06:19:25.565	This request caused a new process to be started for your application, and thus caused your application code to be loaded for the first time. This requ				
+	2012-06-17 06:15:31.450	/resources/facebook/Java	200	12685ms	4kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
I	2012-06-17 06:15:31.450	This request caused a new process to be started for your application, and thus caused your application code to be loaded for the first time. This requ				
+	2012-06-17 06:07:16.650	/resources/facebook/Java	500	17468ms	0kb	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.56 Safari/536.5
E	2012-06-17 06:07:16.589	com.sun.jersey.spi.container.ContainerResponse mapMappableContainerException: The RuntimeException could not be mapped to a response, re-throwing to t				
W	2012-06-17 06:07:16.643	/resources/facebook/Java java.lang.NullPointerException at edu.hm.cc.fbbroker.gae.FacebookBroker.getOutputStream(FacebookBroker.java:55) at sun.reflect.Na				
C	2012-06-17 06:07:16.648	Uncaught exception from servlet.java.lang.NullPointerException at edu.hm.cc.fbbroker.gae.FacebookBroker.getOutputStream(FacebookBroker.java:55) at sun.ref				
I	2012-06-17 06:07:16.650	This request caused a new process to be started for your application, and thus caused your application code to be loaded for the first time. This requ				

Funktionen der Admin-UI: Mehrere Versionen einer Applikation gleichzeitig betreiben.

Version	Default	Deployed	Delete
<input type="radio"/> 1 instances java api_version: 1.0	No	6:48:31 ago by adersberger@googlemail.com	<button>Delete</button>
<input checked="" type="radio"/> 2 instances java api_version: 1.0	Yes	2:19:52 ago by adersberger@googlemail.com	<button>Delete</button>

Make Default

Traffic Splitting

Traffic Splitting is useful for activities such as anonymously testing new features of your application or different versions of your application with a percentage of users. For more information, be sure to checkout [Getting Started with App Engine Traffic Splitting](#).

Running Traffic Splits

Version	Traffic %	Delete
1	<input type="text" value="90"/>	<button>Delete</button>

Add Traffic Split...

Split traffic by: ☒ IP Address ☐ Cookie

Funktionen der Admin-UI: Verwaltung der Plattform-Dienste (z.B. Datastore).

Query

Create

By kind:

KeywordHistoryItem

kinds as of 0:00:08 ago

Options

☐ By GQL:

SELECT * FROM KeywordHistoryItem

Learn more about [GQL syntax](#).

Run Query

KeywordHistoryItem Entities

« Prev 20 1-3 Next 20 »

<input type="checkbox"/> ID/Name	keyword	timestamp	user
<input type="checkbox"/> id=1	Java	2012-06-18 13:09:08.967000	adersberger@googlemail.com
<input type="checkbox"/> id=1001	München	2012-06-18 13:09:24.550000	adersberger@googlemail.com
<input type="checkbox"/> id=2001	Rosenheim	2012-06-18 13:09:38.647000	adersberger@googlemail.com

Delete

« Prev 20 1-3 Next 20 »