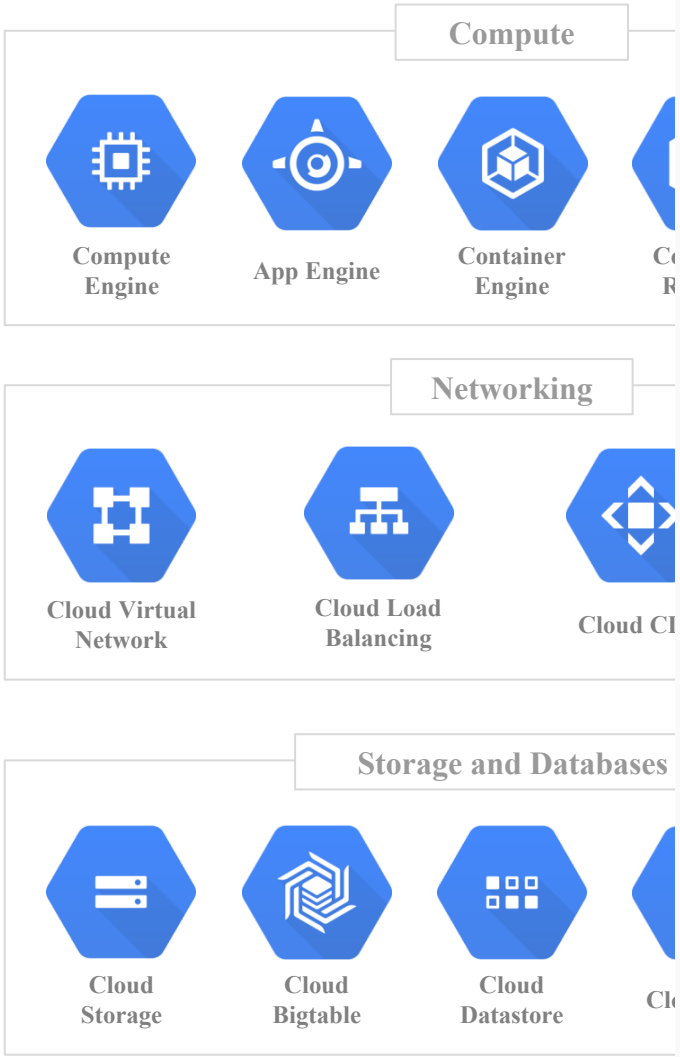
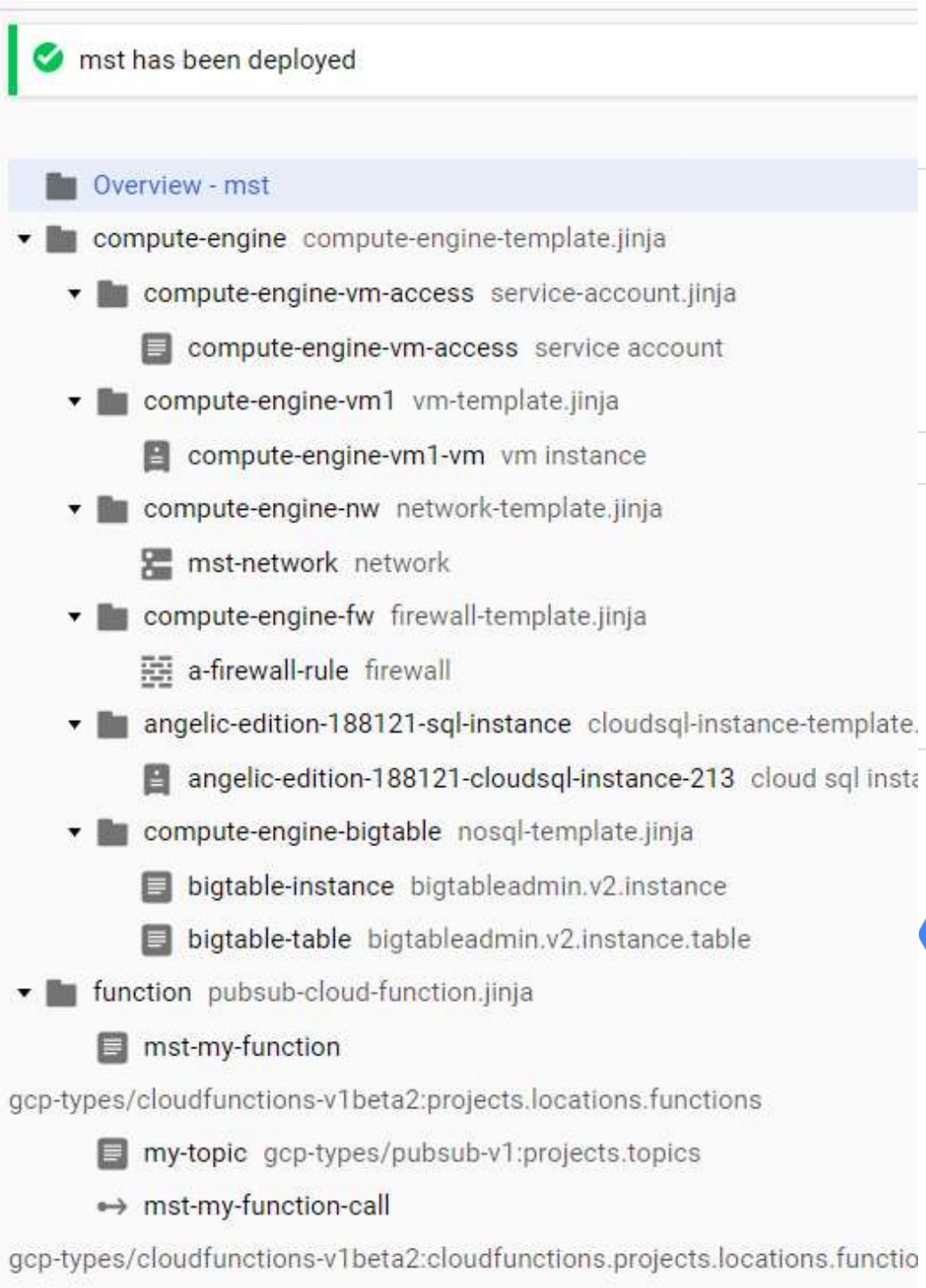
Final Presentation

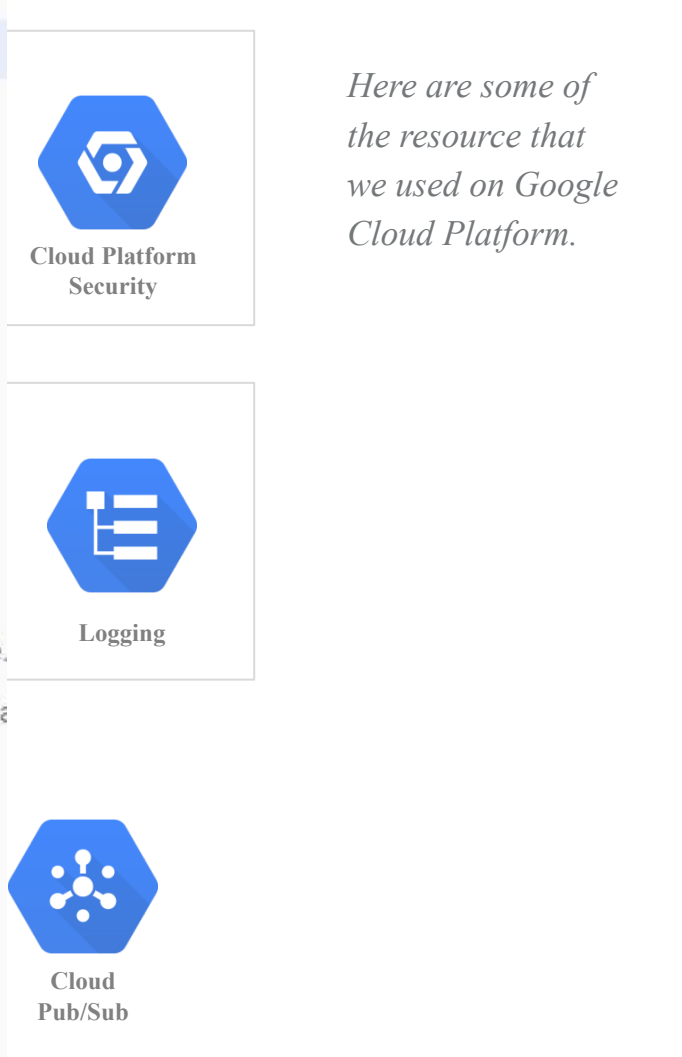
GCP vs AWS

Shirui Wang 001226459 wang.shirui@husky.neu.edu
Wenhe Ma 001238705 ma.wenhe@husky.neu.edu
Yuting Jing 001221590 jing.yu@husky.neu.edu
Haoan Yan 001220895 yan.hao@husky.neu.edu

Google Cloud Platform Service





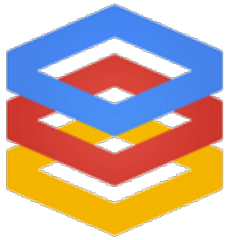


Here are some of the resource that we used on Google Cloud Platform.

ACME Inc.

- Money Cost
- Time Efficiency

Instance



Google Compute Engine

Amazon EC2

	Compute Engine	Amazon EC2
Virtual machines	Instances	Instances
Machine images	Image	Amazon Machine Image
Temporary virtual machines	Preemptible VMs	Spot instances
Firewall	Compute Engine firewall rules	Security groups
Automatic instance scaling	Compute Engine autoscaler	Auto Scaling
Local attached disk	Local SSD	Ephemeral disk
VM import	Supported formats: RAW	Supported formats: RAW, OVA, VMDK, and VHD
Price	f1.micro 0.6G → $0.0096/h	t2.micro 1G → $0.0116/h

Developer Tools

	GCP Deployment Manager	AWS CloudFormation
Deployed collection of resources	Deployment	Stack
Deployment files	Configuration files, template files, and schema files	Template file
Syntax	YAML, Jinja, Python	JSON, YAML
Composition and reuse	Templates	Nested stacks
Identification of individual resources	Name	Logical ID
Scope of deployment locality	Global	Regional
Default maximum number of stacks or deployments	1000	200
Graphical user interface	No	Yes
Preview	Yes	Yes
Stack policies	No	Yes
Price	Free!	

Developer Tools

	GCP Deployment Manager	AWS CloudFormation
Maximum size of template or configuration	1 MB	460.8 KB
Declarative	Yes	Yes
Conditionals	Yes	Yes
Loops	Yes	No
Parameterization	Yes	Yes
Output values	Yes	Yes

IP types

IP type	Cloud Platform	AWS
Permanent IP	Static IP	Elastic IP
Temporary IP	Ephemeral IP	Ephemeral IP
Internal IP	Internal IP	Internal IP

Region

	GCP	AWS
Region	10	15
Zone	30	42
us-east	SC, VA	OH, VA
us-central	IA	
us-west	OR	CA, OR
ca		✓
eu-west	Belgium, London	Ireland, London
eu-central		Frankfrut
asia-east	Taiwan	Beijing
ap-south		Mumbai
…	…	…

AWS has more regions and zones right now.

GCP is catching up fast, in 2018 the gap will be very small.

Network DNS

GCP	AWS
Cloud DNS	Amazon Route 53
✗	Latency-based routing
✗	Geography-based routing
✗	Global Private Zone

Routing

single global IP address -> Simplified DNS setup -> GCP distrubutes load balancer resource in multiple regions.
Geography-based routing is done through load balancer.

Costs

Amazon Route 53 and Cloud DNS both charge based on the number of zones hosted per month and queries per month. Route 53 charges a higher rate for geographic-based routing and latency-based routing queries.

Cloud Pub/Sub

GCP Pub/Sub	AWS SNS
First 10GB $0.00	First 1GB $0.00
Next 50TB $0.06	Next 40TB $0.085
Next 100TB $0.05	Next 100TB $0.07
Beyond 150TB $0.04	Next 350TB $0.050
	No charge for deliveries to Lambda
intended for application and system integrations	direct communication with end-user interfaces

Cloud Functions

	GCP Cloud Functions	AWS Lambda
Code ingestion	Zip upload, IDE, Storage, GitHub	Zip upload, DynamoDB, IDE, S3
Latency	Typically in less than 2 minutes	Typically within seconds
Triggers	Cloud Storage, Pub/Sub	S3, DynamoDB, and SNS
Languages supported	Node.js	Node.js, Java, Python, C#

Virtual Private Cloud

GCP	AWS
Subnets can span region	Subnets constrains to one region
	VPN needed
Consistent ID Zones	Inconsistent ID Zones
…	…

GCP Firewall & AWS Sercurity Group

	GCP FireWall	AWS Security Group
Description	You can create firewall rules to allow or deny specific connections based on a combination of IP addresses, ports, and protocol.	A security group acts as a virtual firewall for your instance to control inbound and outbound traffic.
Differences	- GCP FireWall can specify denied connection, while SecurityGroup can’t - By default, both allow all outbound traffic; GCP deny all ingress traffic	

GCP LoadBalancer & AWS LoadBalancer

GCP LoadBalancer	AWS LoadBalancer
Software-Defined/ Real-Time	VirtualMachines/ Linear
Global Anycast IP	Multiple IPs, No DNS Required
No pre-warming/ 1 Million+ QPS in 15s	Pre-warm/ 30+ min
Automatic multi-region failover, Resilience	Route53 DNS failover, Alternate LBS
Frontends, Backends	Listeners, TargetGroups
Content-Based Routing – L7 (HTTP/S LoadBalancing)	Content-Based Routing – L7 (Application LoadBalancing)

GCP BigTable & GCP DataStore & AWS DynamoDB

	DynamoDB	BigTable	DataStore
Description	Hosted, scalable database service by Amazon with the data stored in Amazons cloud	Google's NoSQL Big Data database service. The same database that powers Google Search, Analytics, Maps, and Gmail.	Automatically scaling NoSQL Database as a Service (DBaaS) on the Google Cloud Platform
Secondary Index	Yes(Restricted)	No	Yes
Data Typing	Yes	No	Yes
SQL	No	No	GQL
API and access method	RESTfull HTTP api	gRPC API, HBase compatible API	gRPC API, RESTful HTTP/JSON API
Map Reduce	No	Yes	Yes
Consistency	Eventual Consistency Immediate Consistency	Immediate Consistency	Immediate Consistency or Eventual Consistency depending on type of query and configuration
Cons.	No Foreign keys	Not good at scaling down	Dynamo has 25x the storage on the free tier

GCP BigTable & GCP DataStore & AWS DynamoDB

	DynamoDB	BigTable	DataStore
Free Tier	- Request: 200 million per month. - Storage: 25 GB - Read: 2.5 million per month - Deploy Global Tables in up to 2 AWS regions.	No free tier	No free tiers, but first few requests of a day is free. - Sotrage: 1 GB - Read: 50,000*100,000 - writes: 20,000*100,000 - deletes: 20,000*100,000
Charges	0.25/GB/month	- Node: 0.65 /node/hr - SSD Storage: 0.17 GB/Month - HDD Storage: 0.026 GB/Month - Network Ingress: Free	Storage: 0.18/GB/Month
Others	- DynamoDB charges more than DataStore in terms of read, write, delete - DynamoDB has better Free Tier than DataStore, but charges more when it’s not		

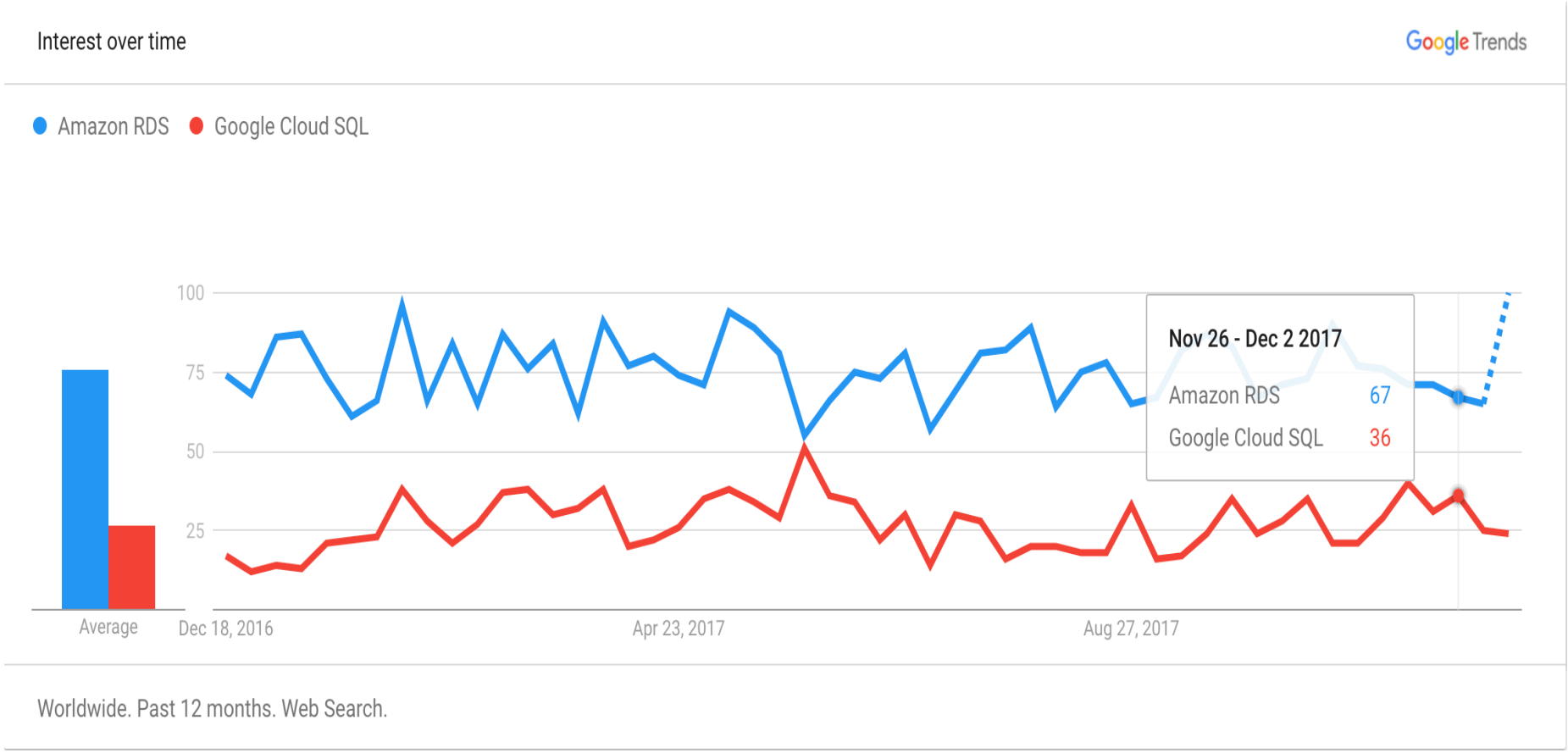
Databases

Cloud SQL

Amazon RDS

	Cloud SQL	RDS
Common	Database service – relational database	
Price	Instance - $0.0350 /h Storage - $0.17 per GB/m	db.t2.micro - $0.017/h
Support Type	MySQL PostgreSQL	MySQL PostgreSQL Aurora MariaDB Oracle Microsoft SQL Server

Databases


Interest over time
Google Trends
Amazon RDS
Google Cloud SQL
100
75
50
25
Average
Dec 18, 2016
Apr 23, 2017
Aug 27, 2017
Nov 26 - Dec 2 2017
Amazon RDS 67
Google Cloud SQL 36
Worldwide. Past 12 months. Web Search.

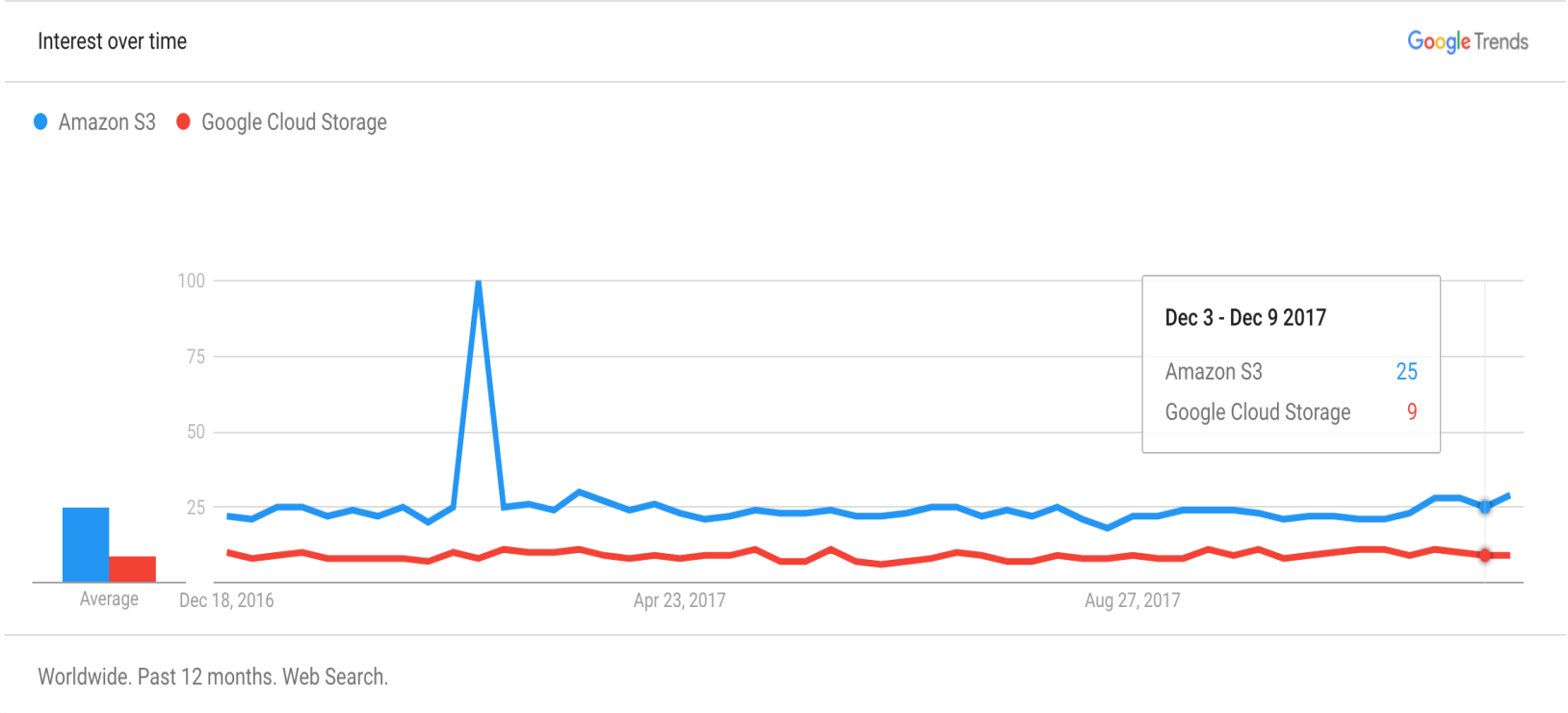
Storage

Cloud Storage

Amazon S3 bucket

	Cloud Storage	S3
Common	Distributed object storage	
Price	$0.026 per GB	$0.023 per GB
Update notifications	Object change notification	Event notification
Deployment locality	Regional Multi-regional	Regional

Storage


Interest over time
Google Trends
Amazon S3
Google Cloud Storage
100
75
50
25
Average
Dec 18, 2016
Apr 23, 2017
Aug 27, 2017
Dec 3 - Dec 9 2017
Amazon S3 25
Google Cloud Storage 9
Worldwide. Past 12 months. Web Search.

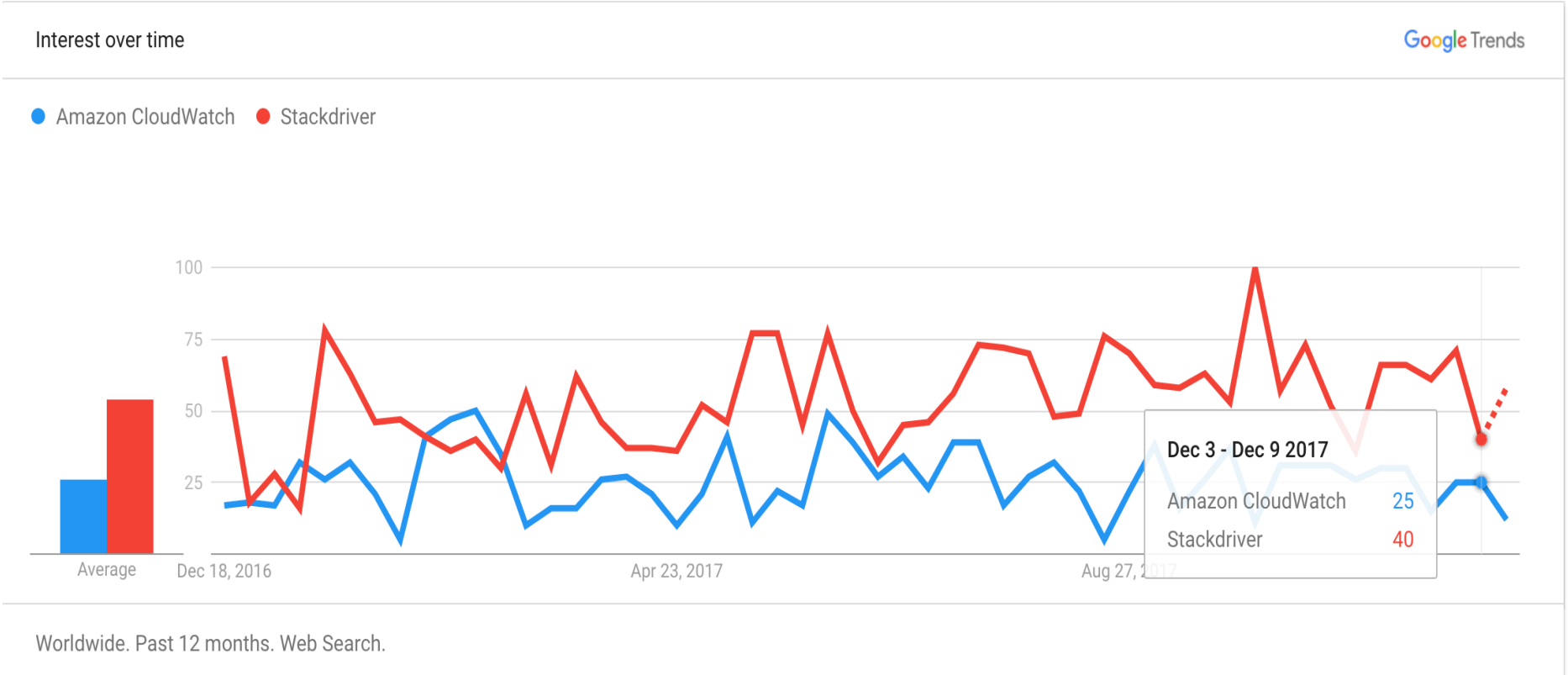
Monitoring

Stackdriver

Amazon CloudWatch

	StackDriver	CloudWatch
Common	Monitor performance, uptime, and overall health of cloud-powered applications.	
Price	free	$3.00 per dashboard /m
Environment	AWS GCP Hybrid	AWS
Function	IT Monitor DevOPs component	IT Monitor

Monitoring


Interest over time
Google Trends
Amazon CloudWatch
Stackdriver
100
75
50
25
Average
Dec 18, 2016
Apr 23, 2017
Aug 27, 2017
Dec 3 - Dec 9 2017
Amazon CloudWatch 25
Stackdriver 40
Worldwide. Past 12 months. Web Search.

Winner

	AWS	GCP
Compute		✓
Storage	✓	✓
Database	✓	
Dev Tools	✓	
Locations	✓	
Monitor		✓

Solution

HYBRID

Thank you

Network DNS

Issue when update DNS using Deploy Manager:

Is there a method to manage creation, updates and deletion of DNS records with DM?
Without the workaround to rename the DM resources.

Thanks

adamharwayne commented on Nov 14 Contributor

Hi,

There is no good way to manage DNS records via DM, because, as you mentioned, each change needs a unique name within DM. In addition, it is not possible for DM to recognize when a record is to be removed.

I put together a simple, non-production proof of concept that seems to work correctly. However, it creates and removes each record on its own, so it is *not* production quality, as it will allow records to disappear for short periods of time before being recreated with a new value (as opposed to the correct behavior of altering the value atomically).

Cloud Pub/Sub

Cloud Functions

```
resources:
- name: {{ env['deployment'] }}-my-function
  type: gcp-types/cloudfunctions-v1beta2:projects.locations.functions
  properties:
    location: {{ properties['region'] }}
    function: my-{{ env['deployment'] }}
    sourceArchiveUrl: {{ properties["sourceArchiveUrl"] }}
    entryPoint: {{ properties['entryPoint'] }}
    eventTrigger:
      resource: $(ref.my-topic.name)
      eventType: providers/cloud.pubsub/eventTypes/topic.publish
- name: my-topic
  type: gcp-types/pubsub-v1:projects.topics
  properties:
    topic: {{ env['deployment'] }}-topic
- name: {{ env['deployment'] }}-my-function-call
  action: gcp-types/cloudfunctions-v1beta2:cloudfunctions.projects.locations.functions.call
  properties:
    name: $(ref.{{ env['deployment'] }}-my-function.name)
    data: |
      {
        "Message": "This is sent by a microservice"
      }
outputs:
- name: cloud-function-response
  value: $(ref.{{ env['deployment'] }}-my-function-call.result)
```