Interacting with Google Cloud Platform

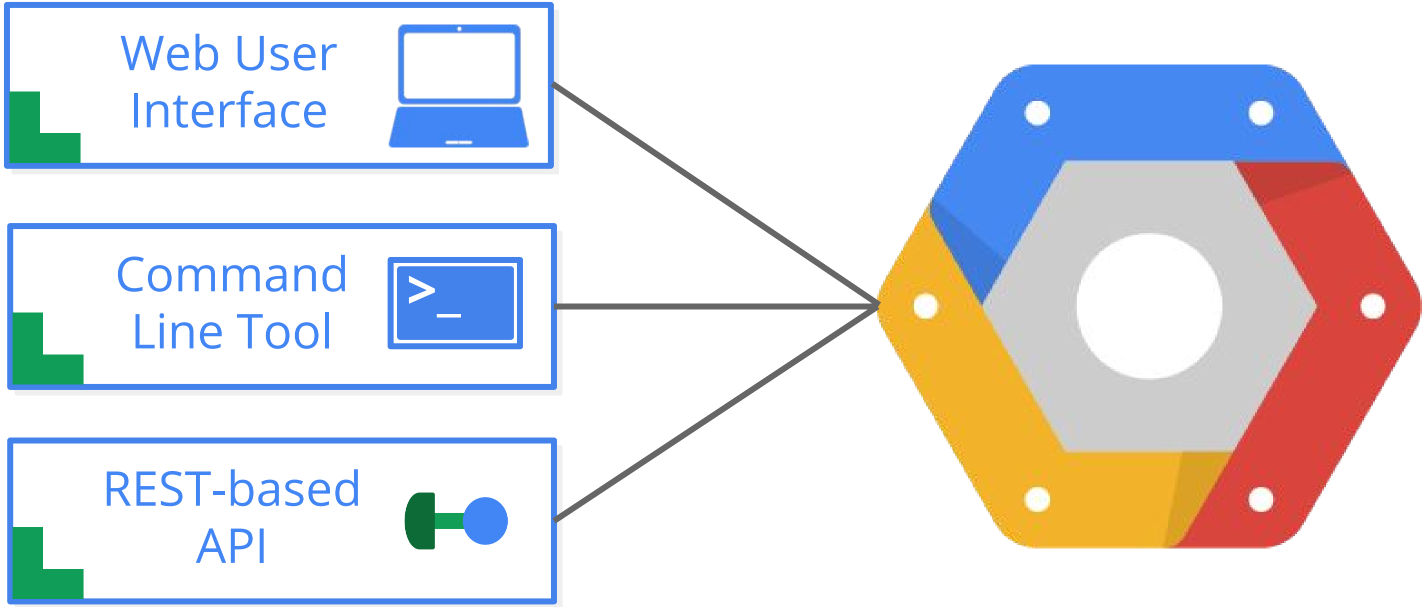
Google Cloud Platform

Google Cloud Platform

Agenda

1. Google Developer Console
2. Cloud SDK
3. Google APIs and API Client Libraries
4. Demo and Lab

Interacting with GCP



Google Developer Console

- Centralized console for all project data
- Manage and create projects including:
  - Billing
  - Project membership
  - API activation
  - OAuth 2.0 credentials
  - Service accounts
  - API keys
- Developer tools
  - Source code repository
  - Monitoring

Agenda

1 Google Developer Console

2 Cloud SDK

3 Google APIs and API Client Libraries

4 Demo and Lab

Cloud SDK

- CLI tools for Linux, OS X and Windows
- Also available as a Docker image
- Install components to manage GCP services
  - **app** for App Engine (preview)
  - **bq** for BigQuery
  - **compute** for Compute Engine
  - **dns** for Cloud DNS
  - **gsutil** for Cloud Storage
  - **sql** for Cloud SQL
- App Engine SDKs are separate downloads (Go, Java, PHP, and Python)
  - https://cloud.google.com/appengine/downloads?hl=en

Agenda

1 Google Developer Console

2 Cloud SDK

3 Google APIs and API Client Libraries

4 Demo and Lab

REST APIs

- Typically use **JSON** as an interchange format
- Some parameters passed in as URL parameters
- Use the **OAuth 2.0** protocol
  - Authentication
  - Authorization
- Enabled through the **Google Developer Console**
- Most APIs include daily **quotas** and **rates** (or limits) that can be raised by request
  - For example Compute Engine uses a default quota of 100 Firewall rules per project
  - Important to **plan ahead** to manage your required capacity

Google API Client Libraries

- Support various languages
  - Java, Python, JavaScript, PHP, .NET
  - Early access for Go, Node.js, Ruby, Dart, Objective-C
- The library handles authorization via OAuth2
  - Provide client ID and client secret

https://developers.google.com/discovery/libraries

Common Google API Client Library Workflow

1. Get user's credentials
   - Run OAuth2 "flow", which interacts between your application, end user and Google Authorization Servers
   - "Flow" may ask user's login and authorization
   - Try it in the OAuth 2.0 Playground and see the breakdown of the "flow"
2. Authorize HTTP object with credentials
   - Internally, sets HTTP header
3. Create service API object with the authorized HTTP object from step 2

https://developers.google.com/accounts/docs/OAuth2

API Client Library Example

Python

```
from apiclient.discovery import build

from oauth2client.client import flow_from_clientsecrets
from oauth2client.file import Storage as CredentialStorage
from oauth2client.tools import run as run_oauth2

flow = flow_from_clientsecrets(CLIENT_SECRETS, scope='https:
//www.googleapis.com/auth/devstorage.full_control',
       message=MISSING_CLIENT_SECRETS_MESSAGE)

credentials = run_oauth2(flow, credential_storage)

http = credentials.authorize(httplib2.Http(timeout=60))


storage = build('storage', 'v1beta2', http=http)

response = storage.objects().list(bucket=bucket_name).execute()
```

define scope

1. run flow

2. authorize HTTP object

3. build service object

use service object

Google Cloud Platform

Agenda

1 Google Developer Console

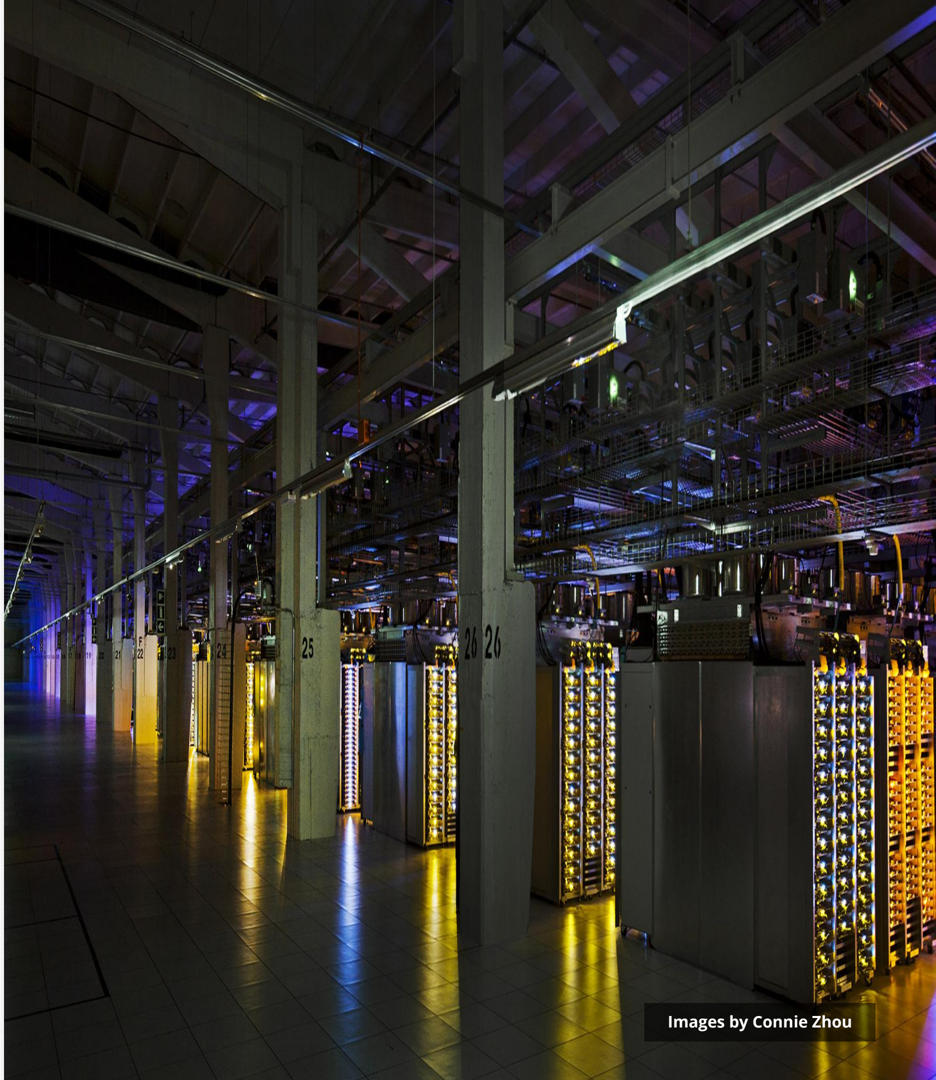
2 Cloud SDK

3 Google APIs and API Client Libraries

4 Demo and Lab

Demo

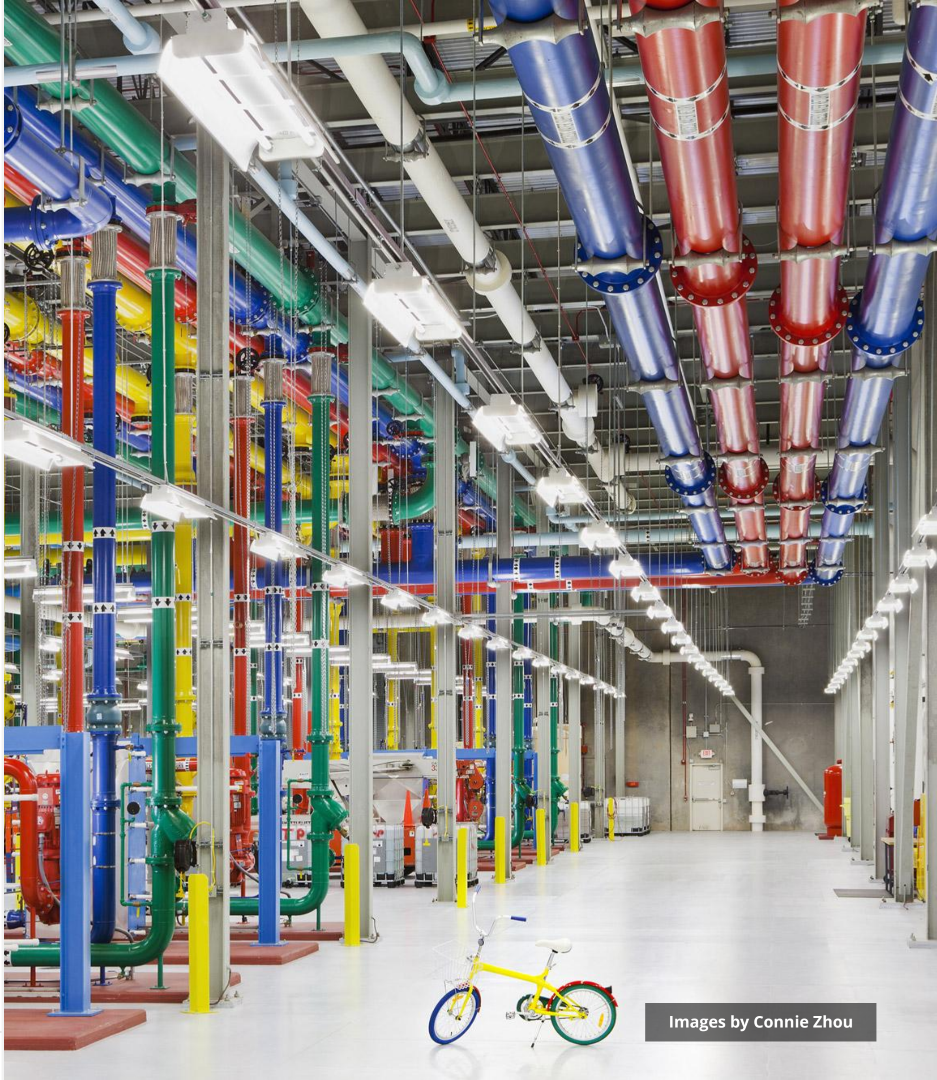
- API Explorer



Images by Connie Zhou

Lab

Configure the Cloud SDK on a Compute Engine instance.



Images by Connie Zhou

Resources

- Cloud SDK installation and quick start
  https://cloud.google.com/sdk/#Quick_Start
- 'gcloud' tool guide
  https://cloud.google.com/sdk/gcloud/
- Google Developer Console
  https://developers.google.com/console/help/new/
- Google API client libraries
  https://developers.google.com/discovery/libraries
- API Explorer
  https://developers.google.com/apis-explorer/

cloud.google.com

Images by Connie Zhou