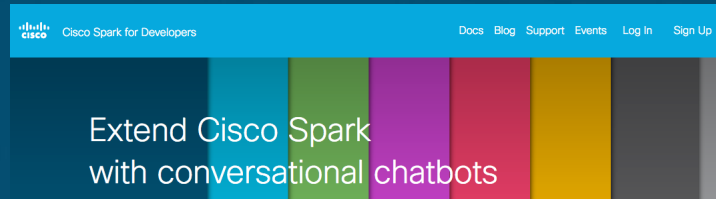


Emerging Technologies Workshops

„Programmierung des Netzwerkes ohne Kenntnisse der Console, ein Traum für Nicht-Cisco-Admins?“

Network Programmability with Cisco APIC-EM





Was sind diese „Emerging Technologies“?

Sind Technologien, die den Status quo verändern können,

sind neu und

umfassen jedoch ältere Technologien, die immer noch umstritten sind und ein relativ geringes Potenzial aufweisen.

frei nach Wikipedia

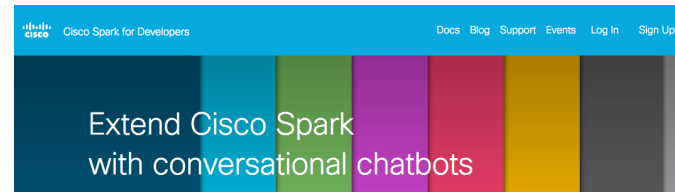


Womit beschäftigen sich diese neuen Kurse / Workshops im Akademieprogramm?



API-Programming (Application Program Interface)

- **Cisco-Spark**
- **APIC-EM**



Python-Programmierung



Kurs- bzw. Workshopinhalte im Überblick?

- *Network Programmability with Cisco APIC-EM*
- *Experimenting with REST APIs using Cisco Spark*

Grundaufbau:

- 2 Kapitel
 - Python-Basics
 - Technologie spezifisch
- Lab-Setup
 - Python installed and verified
 - Postman installed
 - JSONView installed



Kurs- bzw. Workshopinhalte im Überblick?

Was verbirgt sich hinter diesen Workshops?

REST, API-Programming mit Python und Postman

REST (Representational State Transfer)

API-Programming (Application Program Interface)

- Interface zur Anwendungsprogrammierung
- als ein Teil im Betriebssystem, einheitliches Benutzerinterface für Anwendungsprogramme
- offene Schnittstelle für Drittanbieter



REST

Was ist nun REST?

Representational State Transfer oder auch RESTful web services

- API- Typ, der es Programmen erlaubt einfacher Interaktionen über das Internet
- REST API's benutzen HTTP- Aufrufe zwischen den unterschiedlichen Applikationen zum Zugriff auf bzw. zur Manipulation gespeicherter Informationen in Datenbanken.
- Die REST- Architektur ist eine Client / Server –Architektur für ein statless Kommunikationprotokoll.
- Clients und Server tauschen Ressourcen unter Verwendung einer standardisierten Schnittstelle / Protokoll

REST

Was ist nun REST?

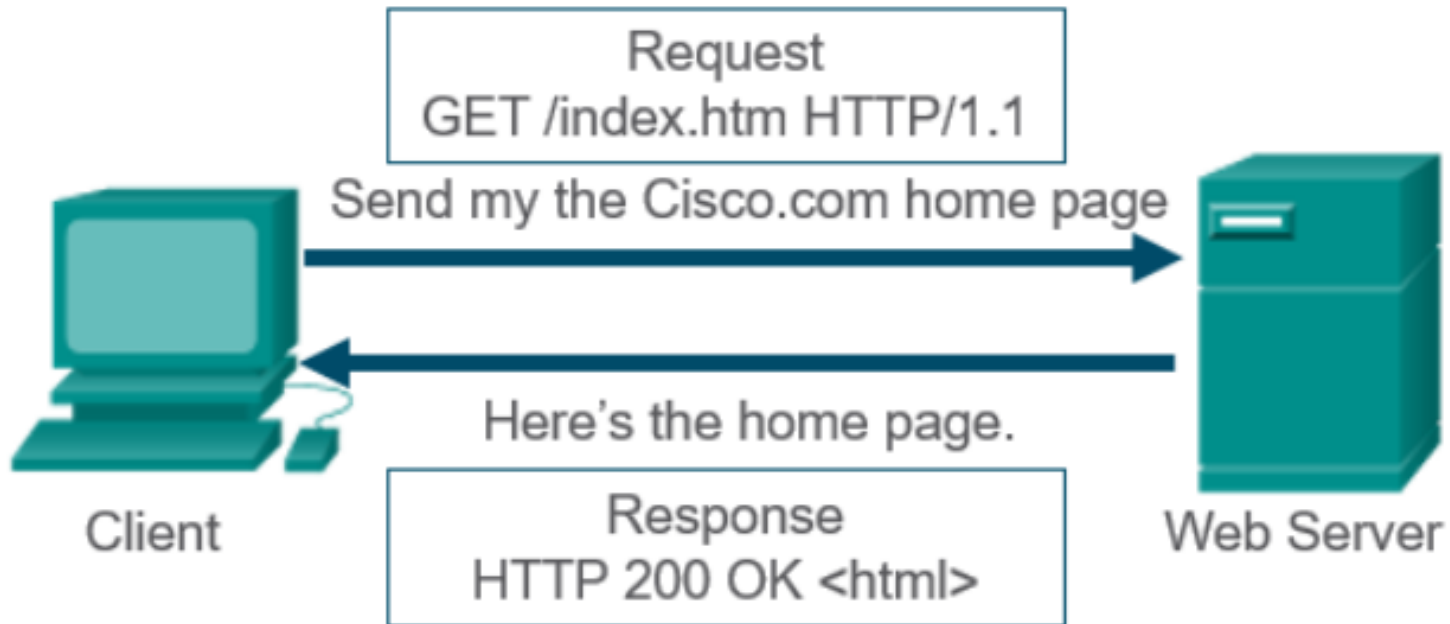
- Benutzt das **HTTP-Protokoll** mit Methoden und Transport
- API- Endpunkte sind ein Server Prozess, der über eine URI angesprochen wird.
- Beschreibung als **“machine-machine-interaction”**
- REST-API ermöglicht Daten-Service zu anderen Herstellern über das Web
- Rückgabeformate in Text-format (XML, JavaScript object notation [JSON])

Method	
GET	<code>https://api.ciscopark.com/v1/people</code>
POST	<code>https://api.ciscopark.com/v1/people</code>
GET	<code>https://api.ciscopark.com/v1/people/{personId}</code>
PUT	<code>https://api.ciscopark.com/v1/people/{personId}</code>
DELETE	<code>https://api.ciscopark.com/v1/people/{personId}</code>
GET	<code>https://api.ciscopark.com/v1/people/me</code>

REST

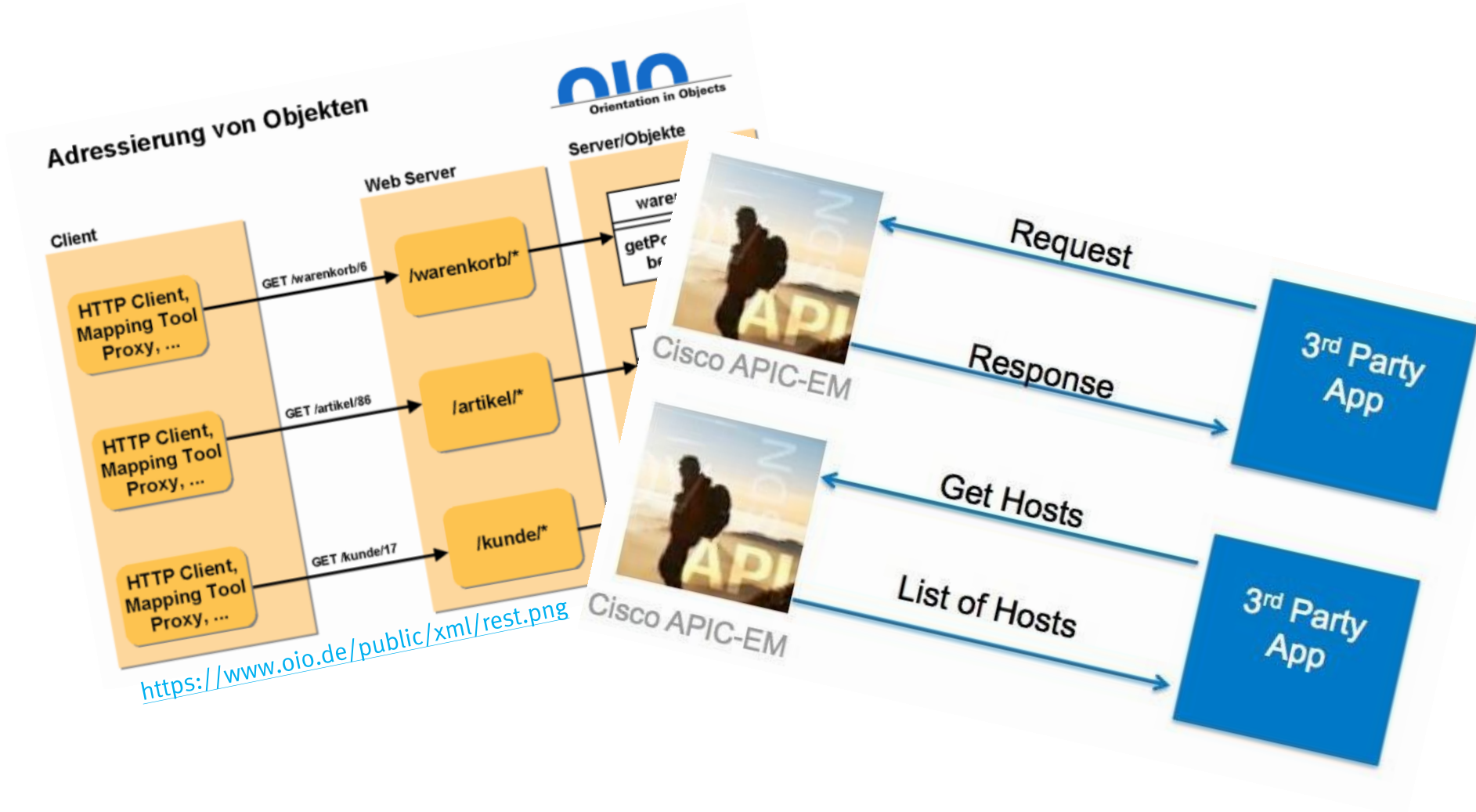
Was ist nun REST?

Normaler HTTP- Aufruf



REST

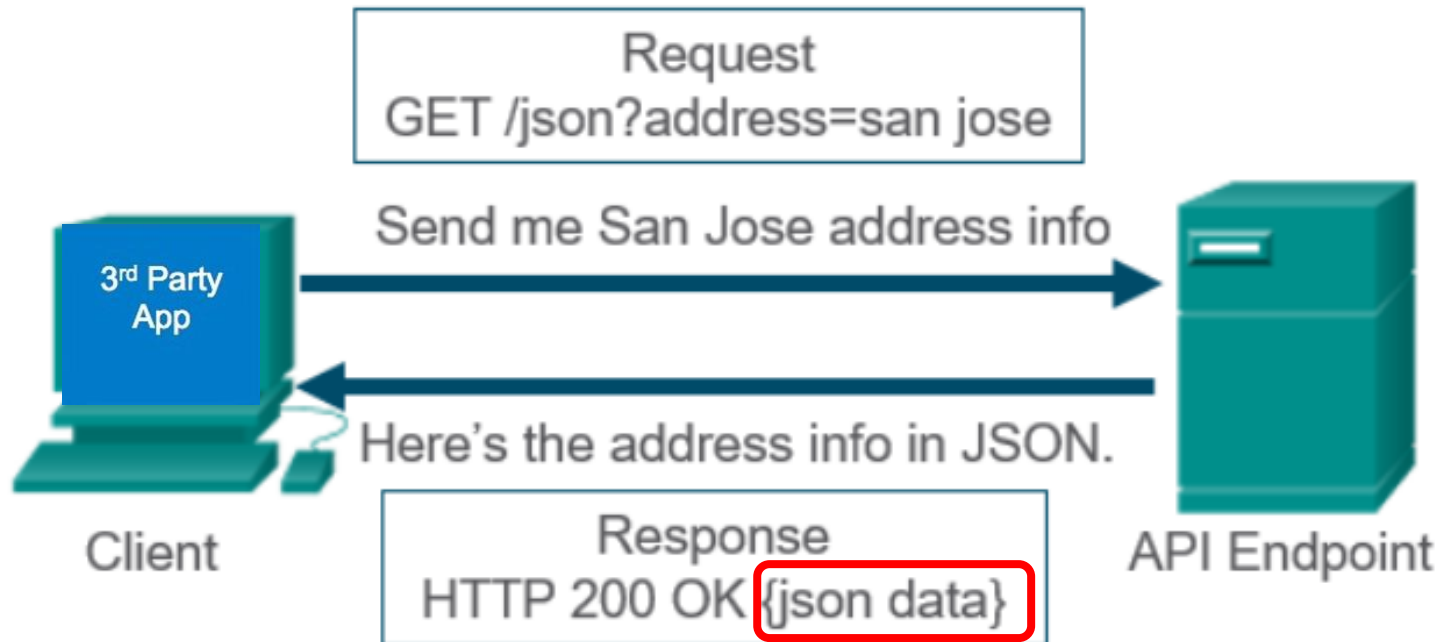
Was ist nun REST?



REST

Was ist nun REST?

Normaler HTTP- Aufruf mit JSON-Rückgabe





REST Was ist nun REST?

Struktur bzw. Anatomie für einen “REST Request”

Method

- GET (retrieve), POST (create), PUT (update), DELETE (remove)

URI

- Example: `http://{APIC-EMController}/api/v1/host`

Authentication

- Basic HTTP, OAuth, none, Custom

Custom Headers

- HTTP Headers
- Example: Content-Type: application/json

Request Body

- JSON or XML containing data needed to complete request



REST

Was ist nun REST?

Methodenbeispiele

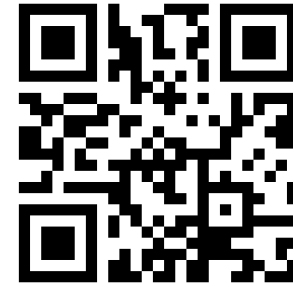
- GET - Rückgabe → allgemeine Informationen wie IOS der Netzwerkgeräte
- POST - Erstellen → Authentikation- Tickets, QoS- Regeln, ACL
- PUT - Update → Update von Policy's – QoS, ACL
- Delete - Löschen → Löschen von Objekten – QoS, ACL

Einige Methoden brauchen dazu separate Authentifizierung.
Bestimmte spezifische Header müssen benutzt werden.
API-Dokumentation beinhaltet diese Spezifizierungen. Lesen!



REST Was ist nun REST?

<https://www.programmableweb.com/apis/directory>



ProgrammableWeb API DIRECTORY API NEWS

LEARN ABOUT APIS WHAT IS AN API? TUTORIALS API CHARTS & RESEARCH

Instantly Create REST APIs from Your Database

- ✓ Connect Your Database
- ✓ Define Security
- ✓ Publish API

Search the Largest API Directory on the Web

Search Over 19,415 APIs

By Category Include Deprecated APIs

API Name	Description	Category	Submitted
Yahoo BOSS	Per provider, this API will be shutting down on March 31, 2016 BOSS (Build your Own Search Service) is Yahoo!'s open search web services platform. The goal of BOSS is...	Search	07.10.2008
Yelp Fusion	The Yelp Fusion APIs are RESTful APIs and users can retrieve business review and rating, information for a particular geographic region or location.display review information for a particular...	Recommendations	08.03.2007
Shopping.com	[This API has been acquired by eBay and now exists as eBay Commerce Network. This profile is being...	eCommerce	01.30.2006

API Name

Google Maps

Twitter

YouTube

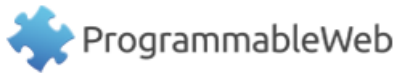
Flickr

Facebook

Amazon Product Advertising



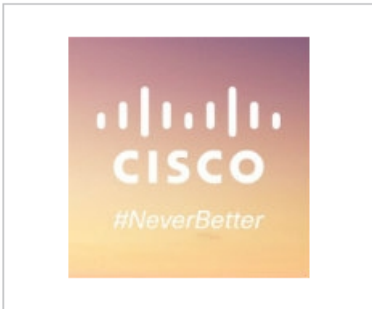
REST Was ist nun REST?



API DIRECTORY ▾ API NEWS ▾

LEARN ABOUT APIS	WHAT IS AN API ?	TUTORIALS	API CHARTS & RESEARCH
------------------	------------------	-----------	-----------------------

Home » APIs » Cisco Spark



Cisco Spark API

Syncing Messaging Teleconferencing

The Cisco Spark API integrates synced conversations through video calls, messaging, and whiteboard architecture with JSON format and OAuth 2.0 virtual room, invite people, search conversations, and notifications from previous conversations.



SPECS	
API Endpoint	https://api.ciscospark.com/v1/
API Portal / Home Page	https://developer.ciscospark.com/getting-started.html
Primary Category	Syncing
Secondary Categories	Messaging, Teleconferencing, Video
API Provider	Cisco
SSL Support	Yes
API Forum / Message Boards	https://developer.ciscospark.com/support.html
Twitter URL	https://twitter.com/ciscosparkdev
Authentication Model	OAuth 1
Version	1
Terms Of Service URL	https://developer.ciscospark.com/tos.html
Is the API Design/Description Non-Proprietary ?	Yes
Type	Web/Internet
Scope	Single purpose API
Device Specific	No
Docs Home Page URL	https://developer.ciscospark.com/getting-started.html
Architectural Style	REST
Supported Request Formats	JSON



Programmierung einer API

Wichtig die Authentifikation!

- **Basic HTTP:** username und password werden in einem String zum Server gesendet!
- **OAuth:** “Open standard for HTTP authentication and session management”
User Token mit entsprechenden Userrechten für den API-Aufruf und Session-Kontrolle.
- **Token:** Ein Token wird gebildet und mit jedem API-Aufruf gesendet. Gilt nur für die Interaktion, kein Managment und Usertracking.



Programmierung einer API

Demo –Aufruf mit REST- APIs!

Struktur des Aufrufes!

<http://maps.googleapis.com/maps/api/geocode/json?address=sanjose>



Demo – Aufruf zum Mitmachen!

<http://maps.googleapis.com/maps/api/geocode/json?address= Fulda>





Demo – Aufruf zum Mitmachen!

<http://maps.googleapis.com/maps/api/geocode/json?address= Fulda>

```
JSON Rohdaten Kopfzeilen
Speichern Kopieren
▼ results:
  ▼ 0:
    ▼ address_components:
      ▼ 0:
        long_name: "Fulda"
        short_name: "Fulda"
        ▼ types:
          0: "locality"
          1: "political"
      ▼ 1:
        long_name: "Fulda"
        short_name: "FD"
        ▼ types:
          0: "administrative_area_level_3"
          1: "political"
      ▼ 2:
        long_name: "Kassel"
        short_name: "KS"
        ▼ types:
          0: "administrative_area_level_2"
          1: "political"
      ▼ 3:
        long_name: "Hessen"
        short_name: "HE"
```

```
JSON Rohdaten Kopfzeilen
Speichern Kopieren
▼ 4:
  long_name: "Deutschland"
  short_name: "DE"
  ▼ types:
    0: "country"
    1: "political"
  formatted_address: "Fulda, Deutschland"
  ▼ geometry:
    ▼ bounds:
      ▼ northeast:
        lat: 50.6293786
        lng: 9.7331007
      ▼ southwest:
        lat: 50.499472399999999
        lng: 9.5659712
    ▼ location:
      lat: 50.5558095
      lng: 9.6808449
```



Programmierung einer API

Was ist mit einer Authentifizierung?

Evtl. muß ein Token von “Google-Developer” benutzt werden!
(Google Account!)

Der Aufruf wird durch diesen Token erweitert!

<http://maps.googleapis.com/maps/api/geocode/json?address=sanjose>





Programmierung einer API

Script zur weiteren Datenverarbeitung in Python.

Aus den JSON- Daten sind weitere Auswertungen möglich (eigene APP)!

Beispiel:

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Apr 17 11:57:22 2018
4
5 @author: Uwe
6 """
7
8 import urllib.parse
9 import requests
10 url = 'http://maps.googleapis.com/maps/api/geocode/json?address= Fulda'
11 json_data = requests.get(url).json()
12 print (json_data)
13 county = json_data['results'][0]['address_components'][3]['long_name']
14 print("\nBundesland:", county)
15 |
```



Programmierung einer API

Script zur weiteren Datenverarbeitung in Python.

Aus den JSON- Daten sind weitere Auswertungen möglich (eigene APP)!

```
Scripts/Phyton-Programme')
{'results': [{ 'address_components': [{ 'long_name': 'Fulda',
'short_name': 'Fulda', 'types': ['locality', 'political']},
{ 'long_name': 'Fulda', 'short_name': 'FD', 'types':
['administrative_area_level_3', 'political']}, { 'long_name':
'Kassel', 'short_name': 'KS', 'types':
['administrative_area_level_2', 'political']}, { 'long_name':
'Hesse', 'short_name': 'HE', 'types':
['administrative_area_level_1', 'political']}, { 'long_name':
'Germany', 'short_name': 'DE', 'types': ['country',
'political']}]}, 'formatted_address': 'Fulda, Germany',
'geometry': { 'bounds': { 'northeast': { 'lat': 50.6293786, 'lng':
9.7331007}, 'southwest': { 'lat': 50.49947239999999, 'lng':
9.5659712}}, 'location': { 'lat': 50.5558095, 'lng': 9.6808449},
'location_type': 'APPROXIMATE', 'viewport': { 'northeast': { 'lat':
50.6293786, 'lng': 9.7331007}, 'southwest': { 'lat':
50.49947239999999, 'lng': 9.5659712}}}}, 'place_id':
'ChIJLbn56Uk0o0cRMKuwKVBDIgQ', 'types': ['locality',
'political']}]}, 'status': 'OK'}
```

Bundesland: Hesse



Warum Python und API's?

	Swagger API Docs	Postman	Python
Availability	Vendor dependent	Yes	Yes
Ease of use	Easy	Medium	More difficult
Save API calls	No	Yes	Yes
Reuse	No	Yes	Yes
Automation	No	No	Yes
Customization	No	No	Yes
App Integration	No	No	Yes

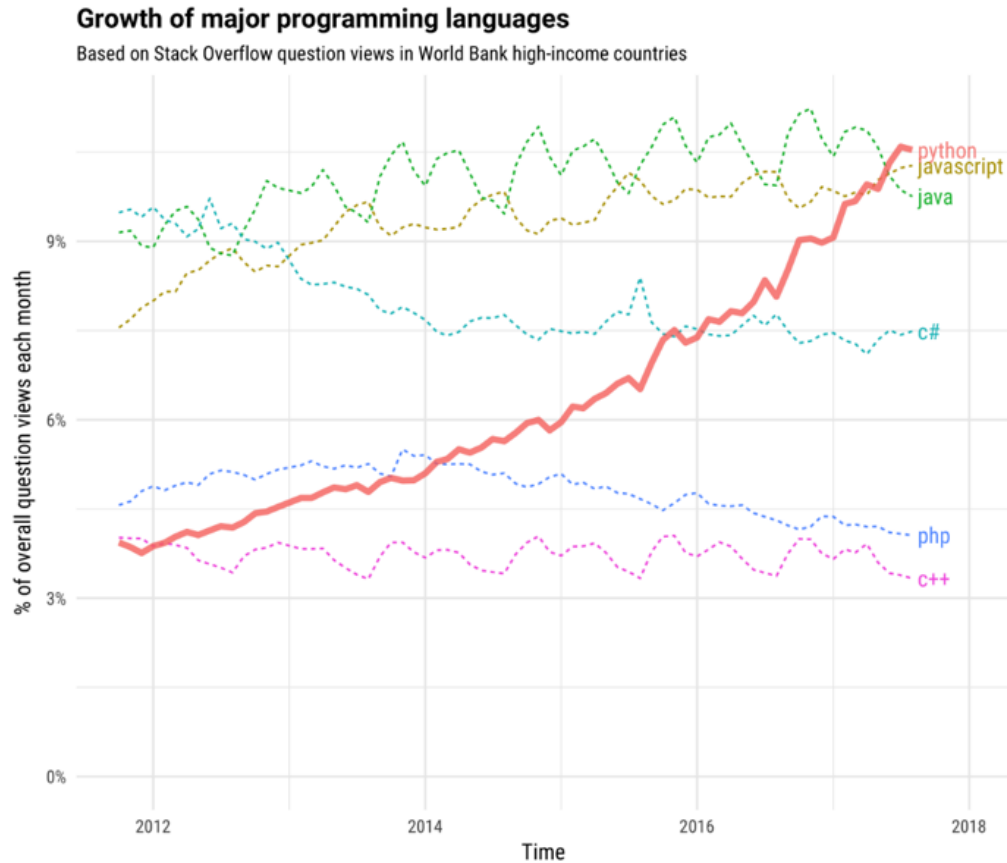


Warum Python und API's?

- **Network Programmability**
- IoT
- Machine Learning
- Artificial Intelligence
- Big Data
- Robotics
- Cybersecurity
- Web Development



Warum Python und API's?



https://stackoverflow.blog/2017/09/06/incredible-growth-python/?utm_source=so-owned&utm_medium=blog&utm_campaign=gen-blog&utm_content=blog-link&utm_term=growth-r



© 2018 Cisco and/or its affiliates. All rights reserved.

For Cisco Networking Academy Use Only.



Communities of Practice (CoPs)

Communities of practice are groups of people who share a concern or a passion for something they do and learn how to do it better as they interact regularly.

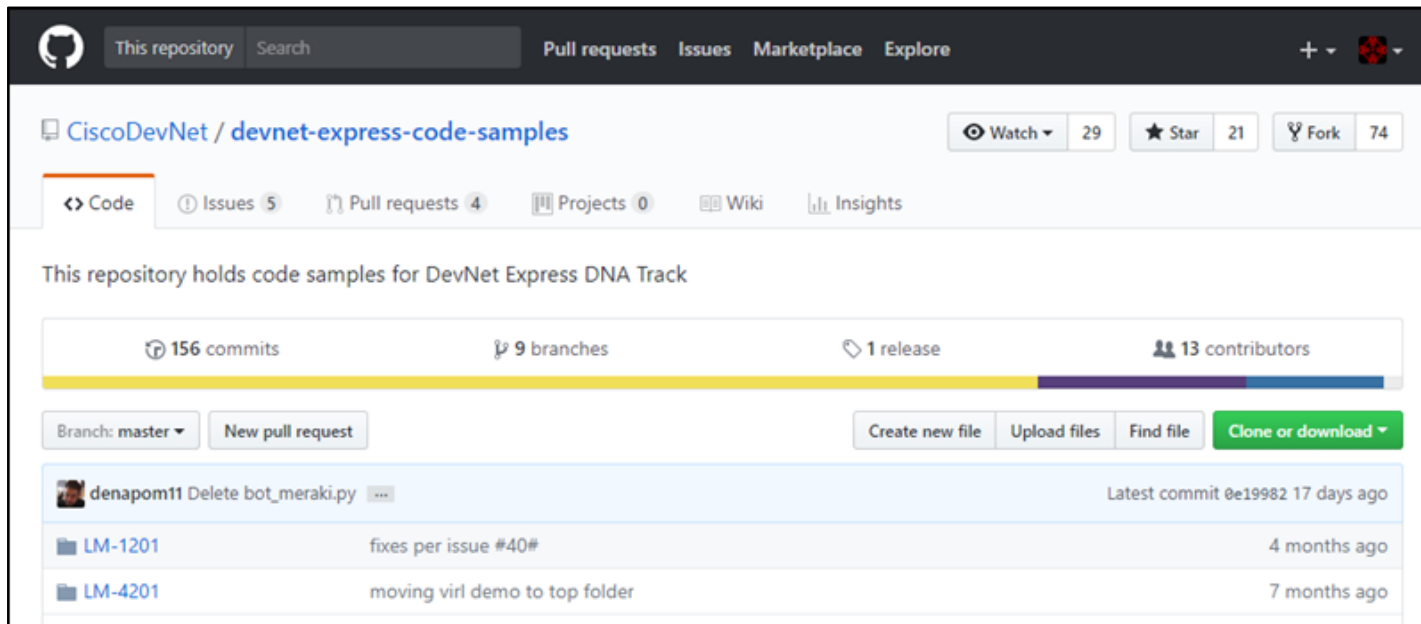
Jean Lave & Etienne Wenger



Warum Python und API's?

CoPs for Programmers - GitHub

GitHub ist “open source software version control system” started by Linus Torvalds, the creator of Linux.



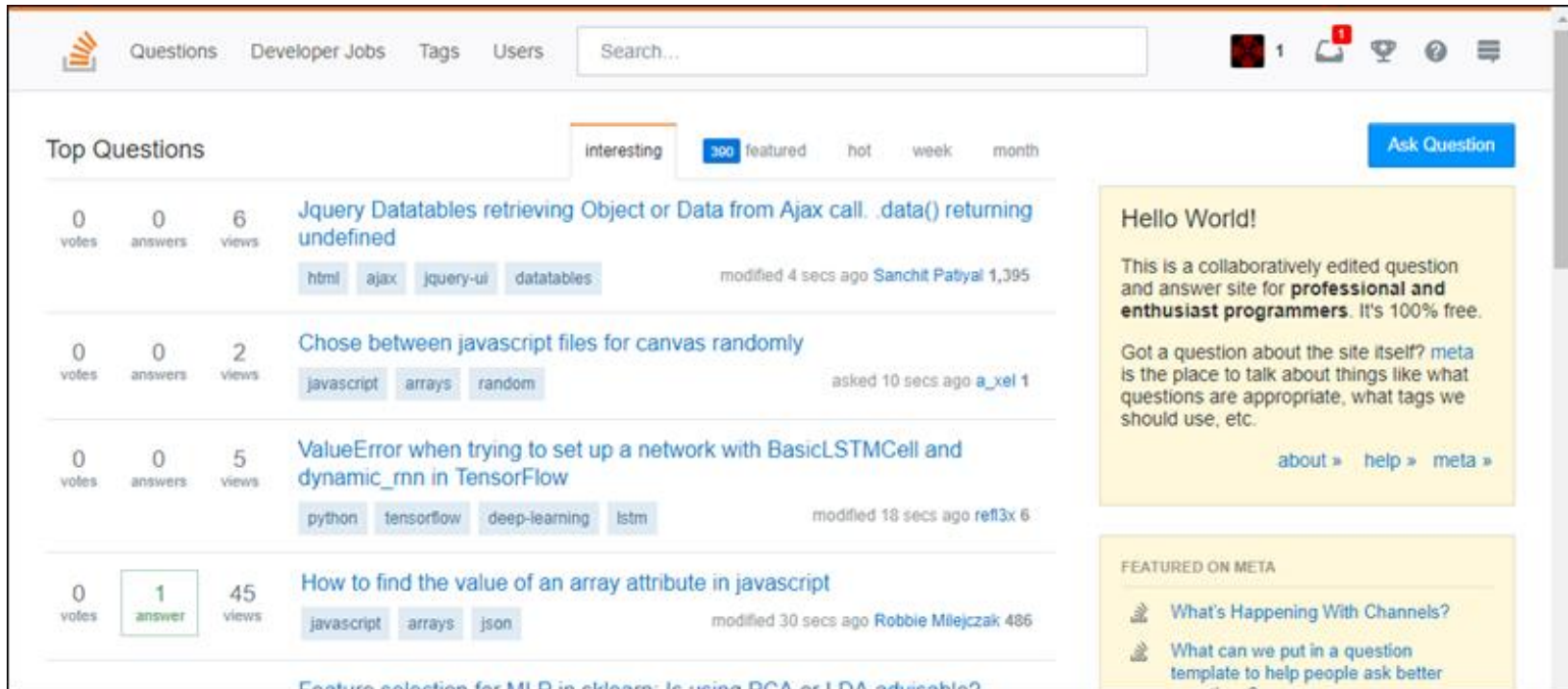
<https://github.com/>



Warum Python und API's?

CoPs for Programmers - Stack Overflow

Stack Overflow ist eine Bibliothek für “jede” Frage über Programmierung.



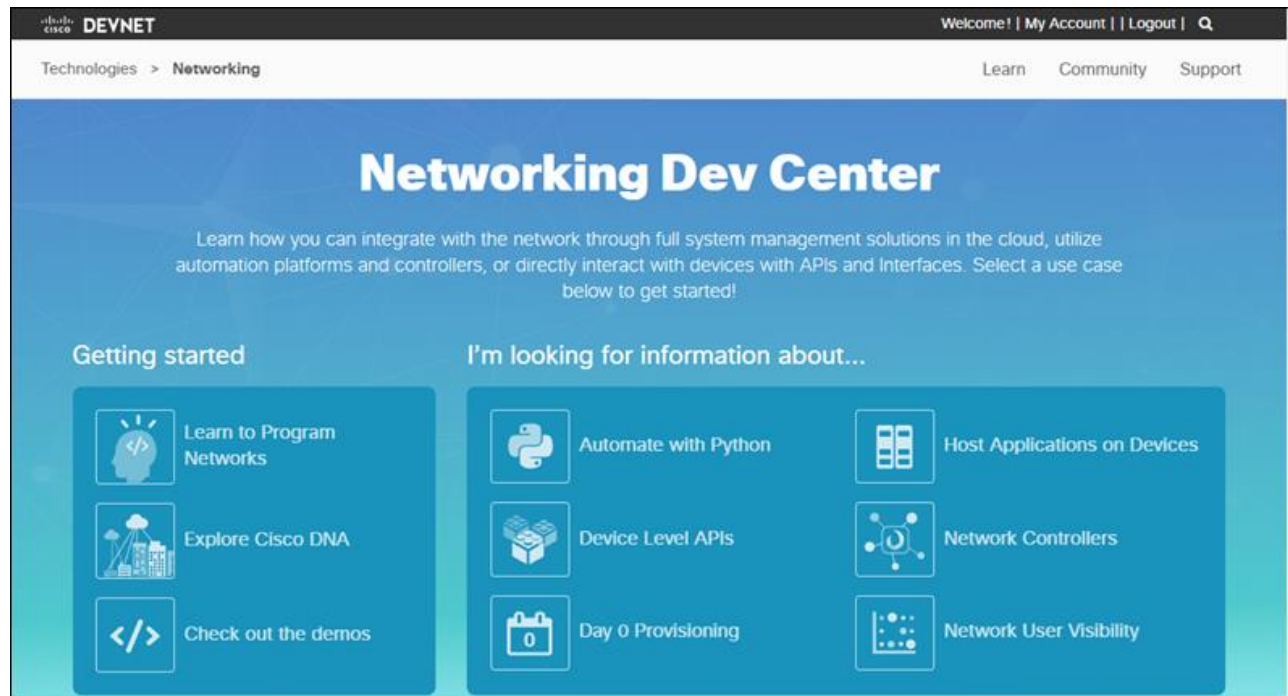
The screenshot shows the Stack Overflow homepage. At the top, there are navigation links for 'Questions', 'Developer Jobs', 'Tags', and 'Users', along with a search bar. The main content area is titled 'Top Questions' and features a list of questions with their respective statistics (votes, answers, views) and tags. The first question is 'Jquery Datatables retrieving Object or Data from Ajax call. data() returning undefined' with 0 votes, 0 answers, and 6 views. The second question is 'Chose between javascript files for canvas randomly' with 0 votes, 0 answers, and 2 views. The third question is 'ValueError when trying to set up a network with BasicLSTMCell and dynamic_rnn in TensorFlow' with 0 votes, 0 answers, and 5 views. The fourth question is 'How to find the value of an array attribute in javascript' with 0 votes, 1 answer, and 45 views. On the right side, there is a 'Hello World!' message and a 'FEATURED ON META' section with two featured questions.



Warum Python und API's?

CoPs for Programmers - Cisco DevNet

Cisco DevNet support Entwickler und Programmierer für Cisco-Anwendungen und Netzwerk-API's!



Postman und API-Programmierung

Ist ein komplettes REST API Entwicklungssystem!



Postman und API-Programmierung

User Interface!

The screenshot displays the Postman interface with a red border. On the left, the 'History' tab is active, showing a list of API calls. A red box highlights the 'Reuse Calls' button. The main workspace shows a GET request to `https://maps.googleapis.com/maps/api/geocode/json?address=güstrow`. The 'Request' section is visible, and the 'Response' section shows the JSON output. The response is a list of address components for 'Güstrow', including 'Güstrow', 'Rostock', and 'Mecklenburg-Vorpommern'.

Reuse Calls

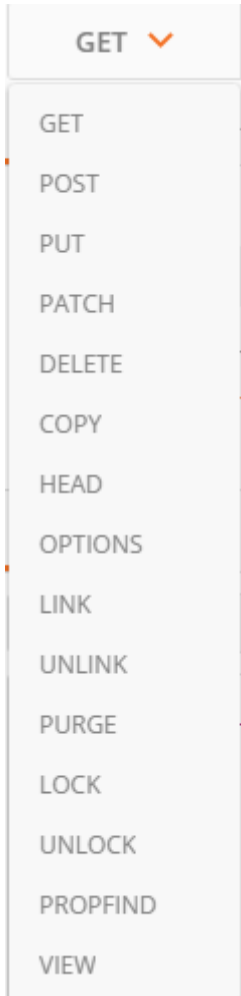
Request

Response

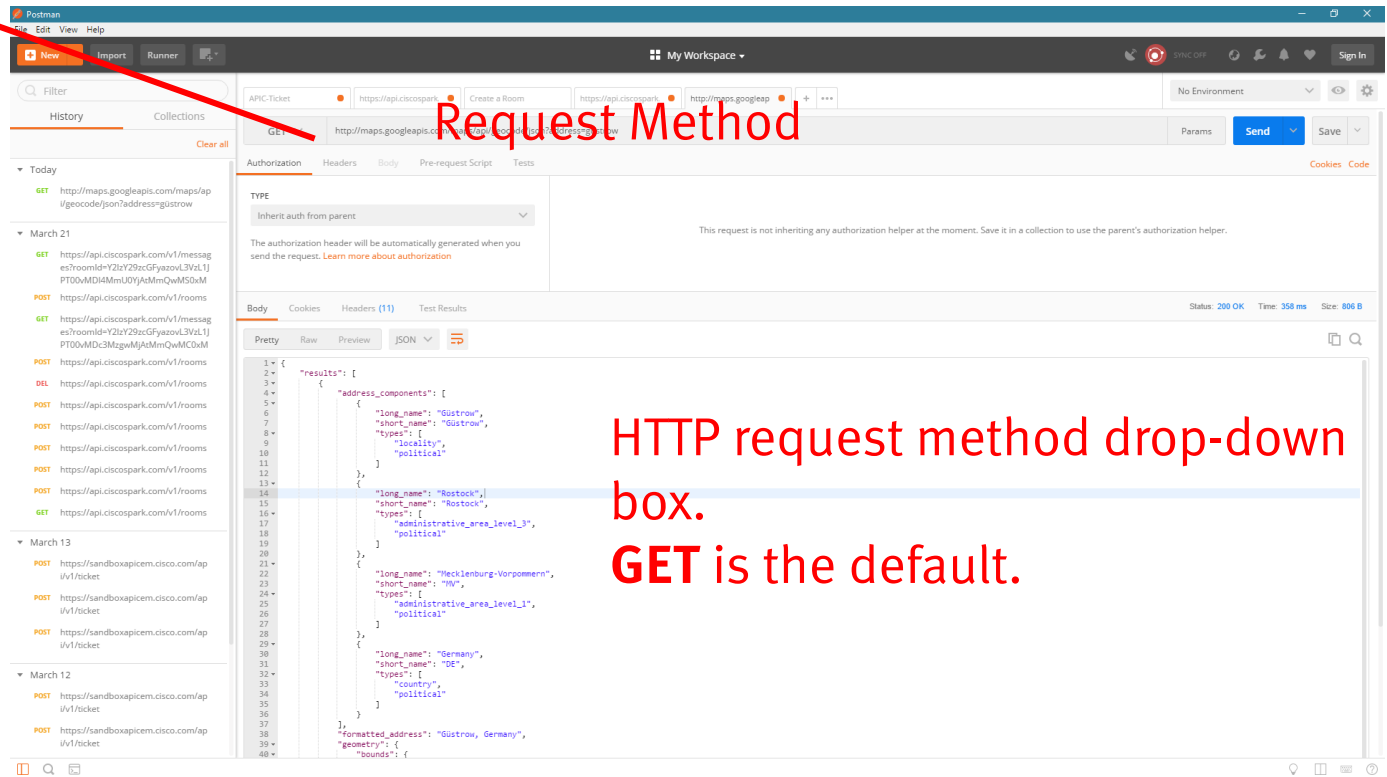
```
1 {
2   "results": [
3     {
4       "address_components": [
5         {
6           "long_name": "Güstrow",
7           "short_name": "Güstrow",
8           "types": [
9             "locality",
10            "political"
11          ]
12        },
13        {
14          "long_name": "Rostock",
15          "short_name": "Rostock",
16          "types": [
17            "administrative_area_level_3",
18            "political"
19          ]
20        },
21        {
22          "long_name": "Mecklenburg-Vorpommern",
23          "short_name": "MV",
24          "types": [
25            "administrative_area_level_1",
26            "political"
27          ]
28        },
29        {
30          "long_name": "Germany",
31          "short_name": "DE",
32          "types": [
33            "country",
34            "political"
35          ]
36        }
37      ],
38      "formatted_address": "Güstrow, Germany",
39      "geometry": {
40        "bounds": {
```

Postman und API-Programmierung

User Interface!



21.04.2019





Postman und API-Programmierung

User Interface!

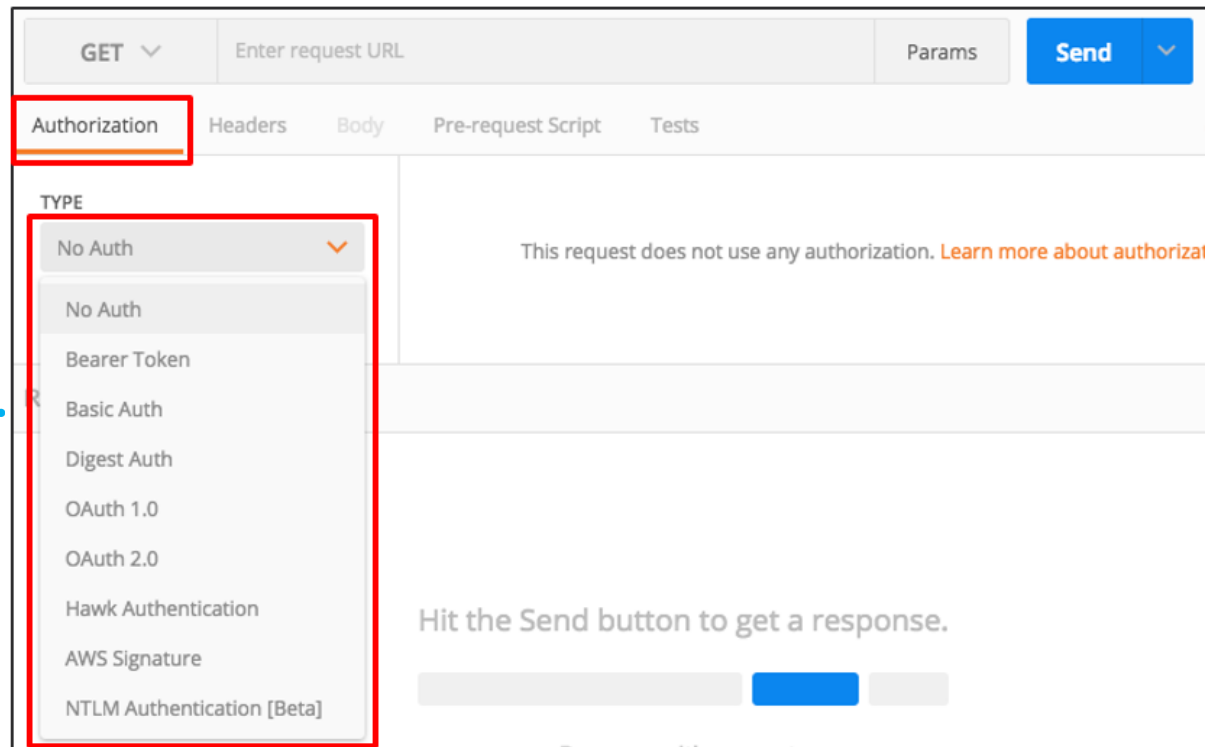
Authorization

Unterschiedliche Möglichkeiten.

Bearer Token

Oauth

Cisco Spark uses **Bearer Token**.



Postman und API-Programmierung

User Interface!

Headers: Content Type

The screenshot shows the Postman interface for a GET request. The 'Headers' tab is selected, and a table of headers is visible. The 'Content-Type' header is checked and its value is 'application/json'. Red boxes highlight the 'Headers (1)' tab, the 'Key' column, and the 'Value' column.

Key	Value	Description	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json				

2. Enter Content-Type key. 3. Enter content-type value.

Content-Type: application/json

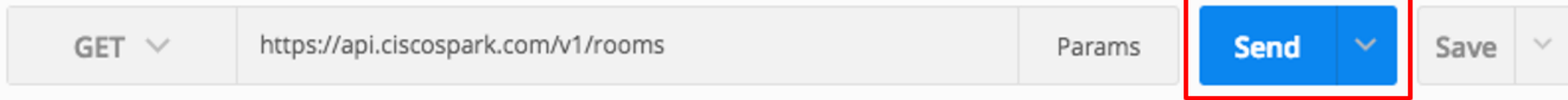
Designates that the content be in JSON format (default)
Postman uses autocomplete for header values and keys.



Postman und API-Programmierung

User Interface

Sending the URL Request

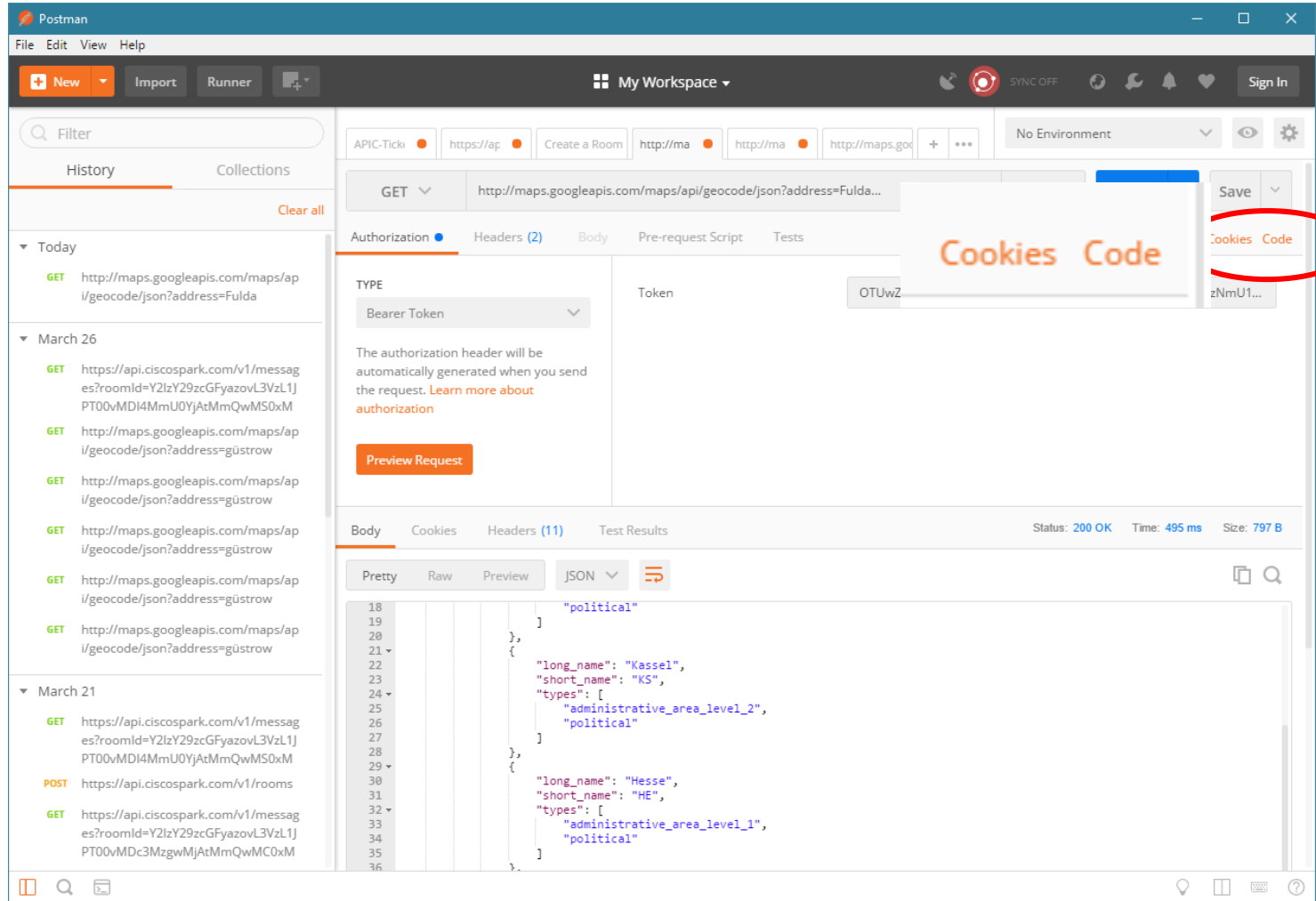


Click **Send** to send the API call.

<http://maps.googleapis.com/maps/api/geocode/json?address=Fulda>

Postman und API-Programmierung

User Interface



Postman und API-Programmierung

User Interface

The screenshot shows the Postman interface with a dropdown menu open on the left, listing various programming languages. The 'Python http.client (Python 3)' option is selected, and a red arrow points to it. The main editor area displays the following Python code snippet:

```
1 import http.client
2
3 conn = http.client.HTTPConnection("maps.googleapis.com")
4
5 headers = {
6     'Authorization': "Bearer OTUwZGEzNDItMmVlZC00Njk0LWFjMzYtMGE5MjA3NTQ4ZjQzNmU1NzJmYjktYmE1"
7     ,
8     'Cache-Control': "no-cache",
9     'Postman-Token': "87c7f343-87a7-4730-80df-30acc477a360"
10 }
11 conn.request("GET", "maps/api/geocode/json", headers=headers)
12
13 res = conn.getresponse()
14 data = res.read()
15
16 print(data.decode("utf-8"))
```

A 'Copy to Clipboard' button is visible in the top right corner of the code editor area.



Kurs- bzw. Workshopinhalte im Überblick?

Experimenting with REST APIs using Cisco Spark

Grundaufbau:

- 2 Kapitel
 - Python-Basics
 - Technologie Spark spezifisch
- Lab-Setup
 - Python installed and verified
 - Postman installed
 - JSONView installed

Registered at Cisco Spark for Developers:
<https://developer.ciscospark.com/>
Cisco Spark client installed.

Experimenting with REST APIs using Cisco Spark

Was ist Spark?

Eine cloud-basierendes “collaboration suite” :

- Meet
- Message
- Call
- Whiteboard
- Share

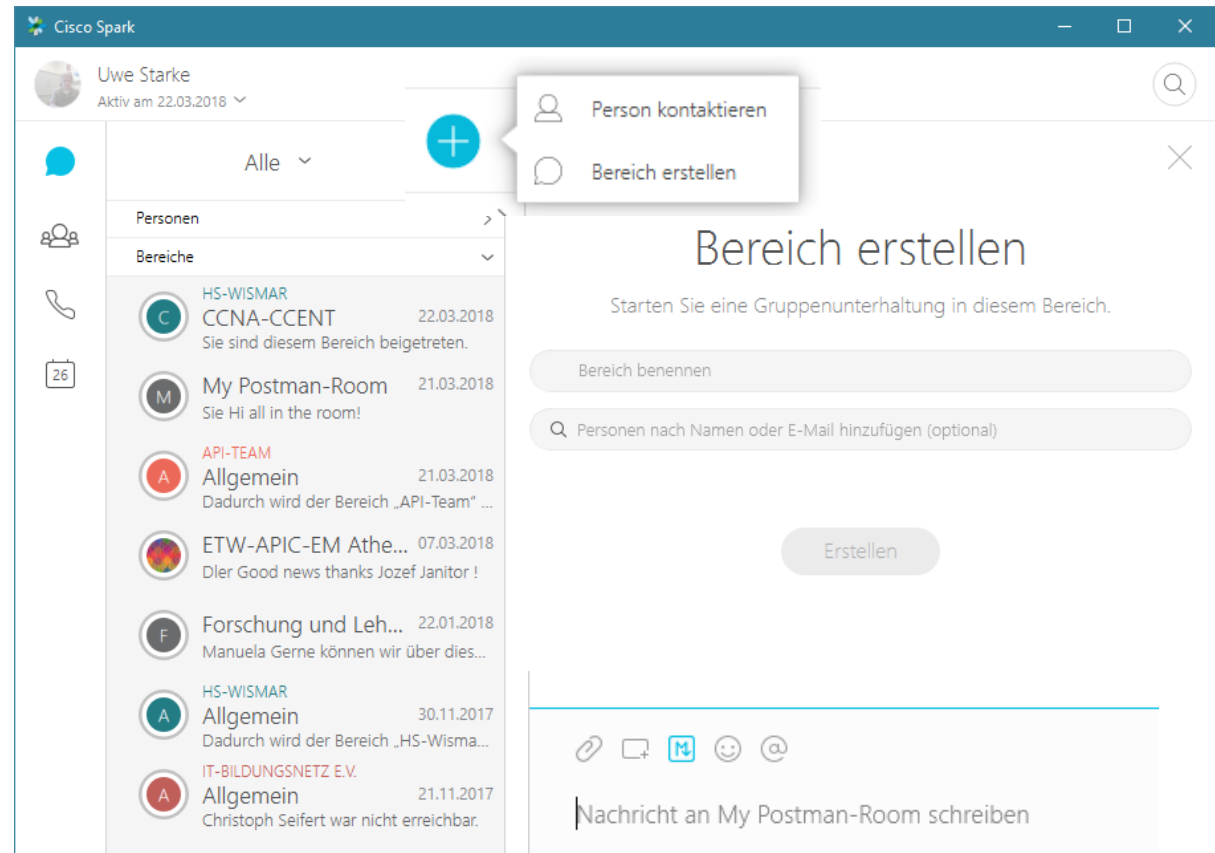


<https://www.youtube.com/watch?v=OavRkK8ir4>

Experimenting with REST APIs using Cisco Spark

Was ist Spark?

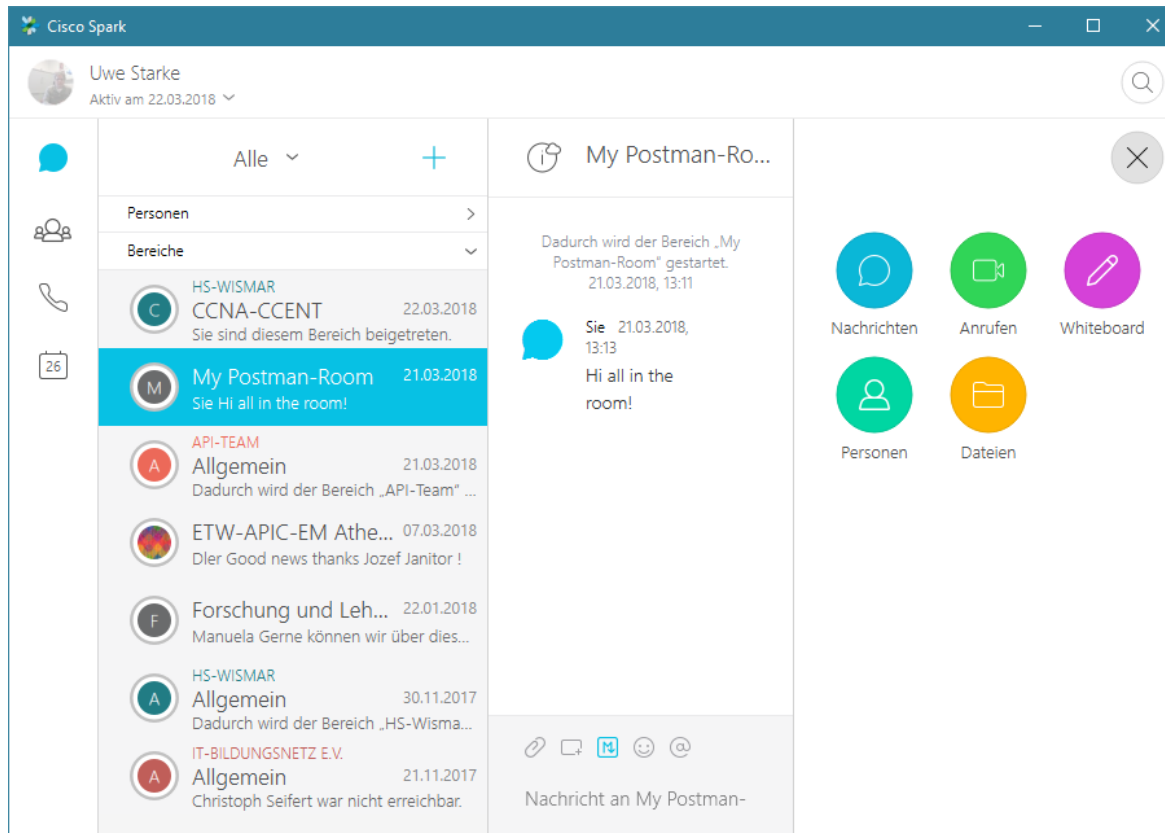
- People join spaces to meet, message,
- and share with other people.
- The first step is to create or join a space.
- Use the “+” sign to add a person or to create a space.
- Select your room and send your message.





Experimenting with REST APIs using Cisco Spark

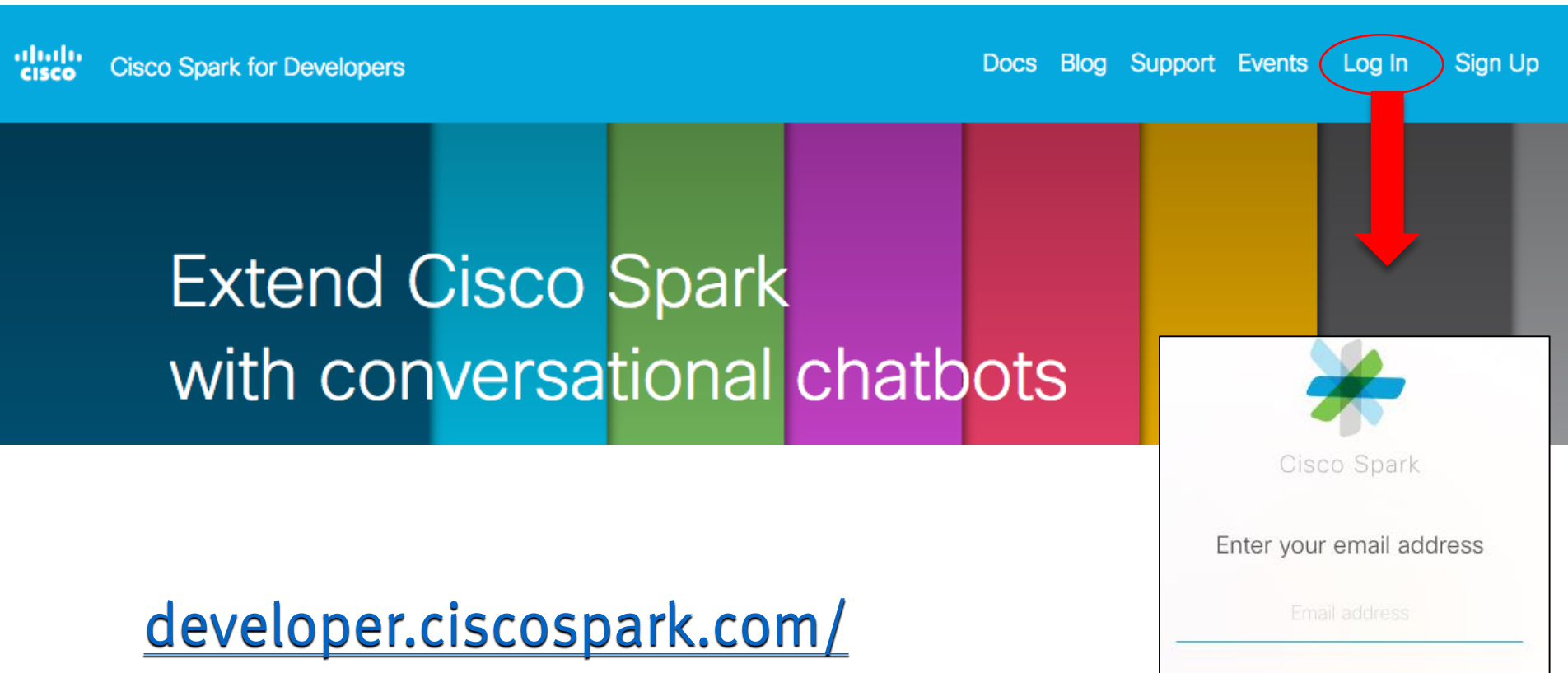
Was ist Spark?





Experimenting with REST APIs using Cisco Spark

Cisco Spark for Developers



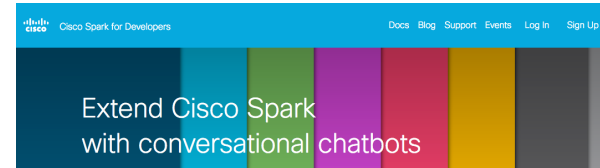
The screenshot shows the top navigation bar of the Cisco Spark for Developers website. The 'Log In' button is circled in red, and a red arrow points from it to a login form. The form contains the Cisco Spark logo, the text 'Cisco Spark', the prompt 'Enter your email address', and an input field labeled 'Email address'.

developer.ciscospark.com/



Experimenting with REST APIs using Cisco Spark

Cisco Spark for Developers



- Create an account
- Play with Spark

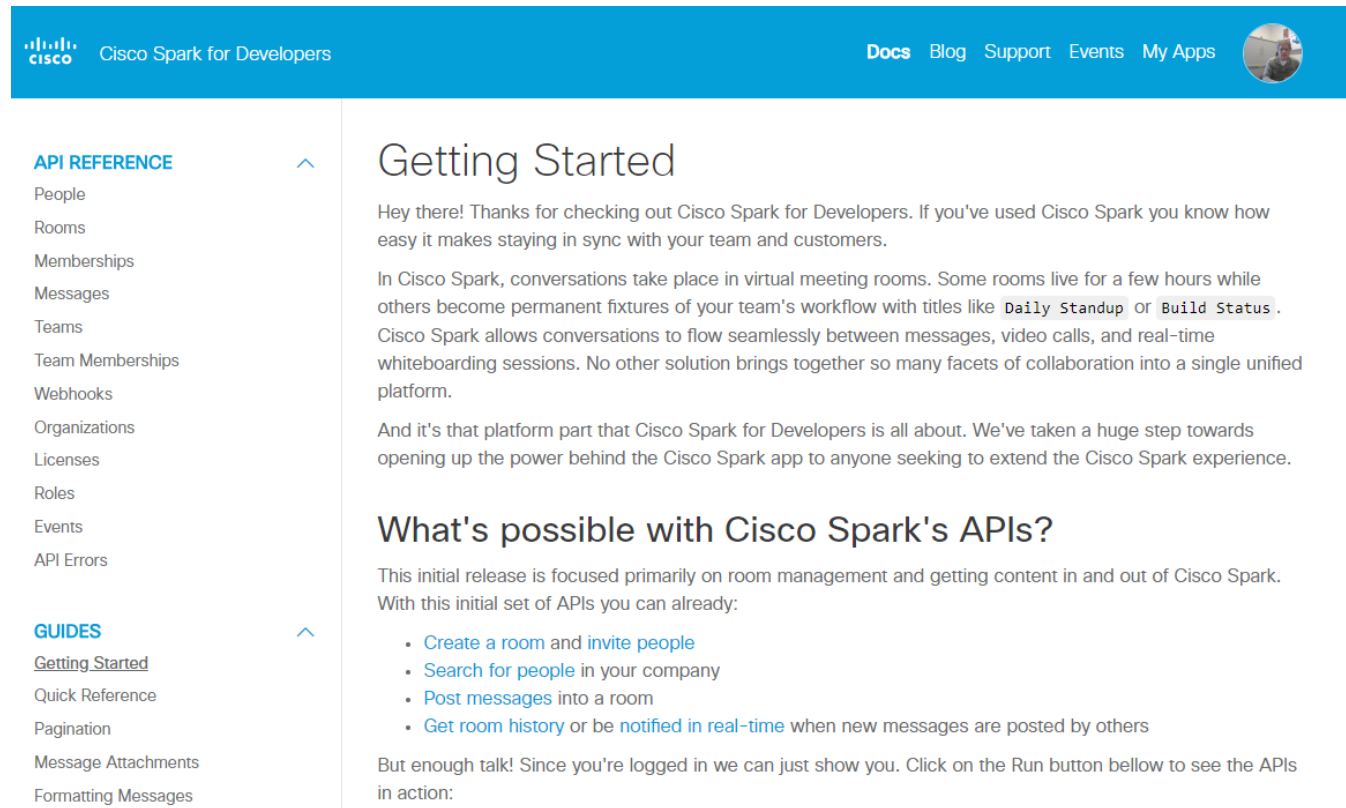
www.ciscospark.com

In this demo and lab, you will be introduced to Cisco Spark and Cisco Spark for Developers. You will learn how to obtain your Cisco Spark Access Token which will be used in later labs.



Experimenting with REST APIs using Cisco Spark

API Reference



The screenshot shows the Cisco Spark for Developers API Reference page. The header is blue with the Cisco logo and navigation links: Docs, Blog, Support, Events, My Apps, and a user profile picture. The left sidebar is white and contains a list of API reference categories under 'API REFERENCE' and 'GUIDES'. The main content area is white and features the 'Getting Started' article, which includes an introduction, a paragraph about virtual meeting rooms, and a list of possible actions with the APIs.

API REFERENCE ^

- People
- Rooms
- Memberships
- Messages
- Teams
- Team Memberships
- Webhooks
- Organizations
- Licenses
- Roles
- Events
- API Errors

GUIDES ^

- [Getting Started](#)
- Quick Reference
- Pagination
- Message Attachments
- Formatting Messages

Getting Started

Hey there! Thanks for checking out Cisco Spark for Developers. If you've used Cisco Spark you know how easy it makes staying in sync with your team and customers.

In Cisco Spark, conversations take place in virtual meeting rooms. Some rooms live for a few hours while others become permanent fixtures of your team's workflow with titles like **Daily Standup** or **Build Status**. Cisco Spark allows conversations to flow seamlessly between messages, video calls, and real-time whiteboarding sessions. No other solution brings together so many facets of collaboration into a single unified platform.

And it's that platform part that Cisco Spark for Developers is all about. We've taken a huge step towards opening up the power behind the Cisco Spark app to anyone seeking to extend the Cisco Spark experience.

What's possible with Cisco Spark's APIs?

This initial release is focused primarily on room management and getting content in and out of Cisco Spark. With this initial set of APIs you can already:

- [Create a room](#) and [invite people](#)
- [Search for people](#) in your company
- [Post messages](#) into a room
- [Get room history](#) or be [notified in real-time](#) when new messages are posted by others

But enough talk! Since you're logged in we can just show you. Click on the Run button below to see the APIs in action:



Experimenting with REST APIs using Cisco Spark

The Rooms API

This API lets you manage Spark spaces (rooms).

Each API endpoint has a REST method.

API REFERENCE

- People
- Rooms**
- List Rooms
- Create a Room
- Get Room Details
- Update a Room
- Delete a Room
- Memberships
- Messages
- Teams
- Team Memberships
- Webhooks
- Organizations
- Licenses
- Roles
- Events
- Metrics

Rooms

Rooms are virtual meeting places where people post messages and collaborate to get work done. This API is used to manage the rooms themselves. Rooms are created and deleted with this API. You can also update a room to change its title, for example.

To create a team room, specify the a `teamId` in POST payload. Note that once a room is added to a team, it cannot be moved. To learn more about managing teams, see the [Teams API](#).

To manage people in a room see the [Memberships API](#).

To post content see the [Messages API](#).

Method	Description
GET	https://api.ciscospark.com/v1/rooms List Rooms
POST	https://api.ciscospark.com/v1/rooms Create a Room
GET	https://api.ciscospark.com/v1/rooms/{roomId} Get Room Details
PUT	https://api.ciscospark.com/v1/rooms/{roomId} Update a Room
DELETE	https://api.ciscospark.com/v1/rooms/{roomId} Delete a Room

Experimenting with REST APIs using Cisco Spark

The Team Memberships API

API REFERENCE ^

- People
- Rooms
- Memberships**
- Messages
- Teams

Team Memberships

- List Team Memberships
- Create a Team Membership
- Get Team Membership Details
- Update a Team Membership
- Delete a Team Membership

Team Memberships

Team Memberships represent a person's relationship to a team. Use this API to list members of any team that you're in or create memberships to invite someone to a team. Team memberships can also be updated to make someone a moderator or deleted to remove them from the team. Just like in the Spark app, you must be a member of the team in order to list its memberships or invite people.

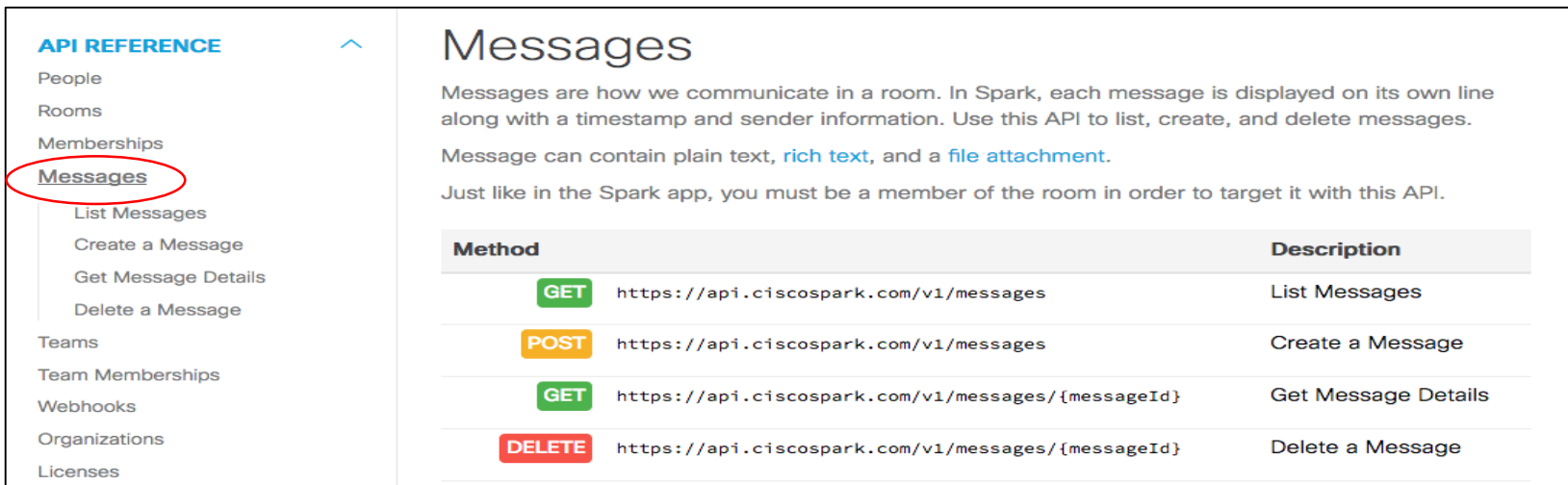
Method	Description
GET https://api.ciscospark.com/v1/team/memberships	List Team Memberships
POST https://api.ciscospark.com/v1/team/memberships	Create a Team Membership
GET https://api.ciscospark.com/v1/team/memberships/{membershipId}	Get Team Membership Details
PUT https://api.ciscospark.com/v1/team/memberships/{membershipId}	Update a Team Membership
DELETE https://api.ciscospark.com/v1/team/memberships/{membershipId}	Delete a Team Membership

- Memberships represent a person's relationship to the room.
- This API lets you manage these memberships.



Experimenting with REST APIs using Cisco Spark

The Messages API



The screenshot shows the Cisco Spark API Reference page for the Messages API. On the left, a sidebar lists various API categories, with 'Messages' circled in red. The main content area is titled 'Messages' and includes a brief description of how messages are used in Spark, followed by a table of API endpoints.

API REFERENCE

- People
- Rooms
- Memberships
- Messages**
- List Messages
- Create a Message
- Get Message Details
- Delete a Message
- Teams
- Team Memberships
- Webhooks
- Organizations
- Licenses

Messages

Messages are how we communicate in a room. In Spark, each message is displayed on its own line along with a timestamp and sender information. Use this API to list, create, and delete messages.

Message can contain plain text, [rich text](#), and a [file attachment](#).

Just like in the Spark app, you must be a member of the room in order to target it with this API.

Method		Description
GET	https://api.ciscospark.com/v1/messages	List Messages
POST	https://api.ciscospark.com/v1/messages	Create a Message
GET	https://api.ciscospark.com/v1/messages/{messageId}	Get Message Details
DELETE	https://api.ciscospark.com/v1/messages/{messageId}	Delete a Message

Messages are how we communicate in a room.
This API lets you list, delete, and create messages.



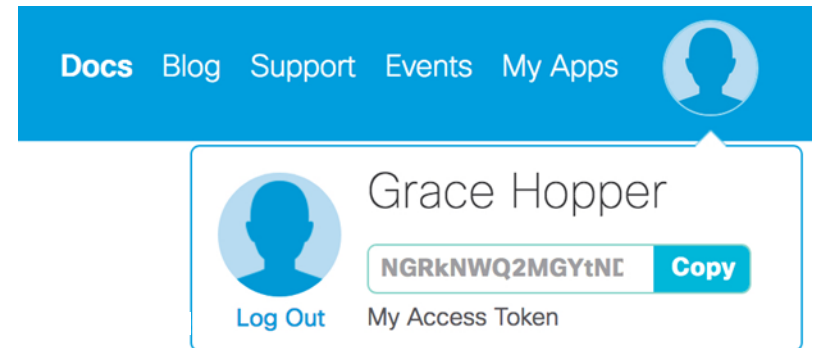
Experimenting with REST APIs using Cisco Spark

The Access Token

Using Cisco Spark APIs requires an Access Token.

The Access Token:

- Is automatically available to you when logged in at the Cisco Spark Developer site.
- Expires in 2 years or **is inactivated when you log out** from Cisco Spark for Developers



Experimenting with REST APIs using Cisco Spark

The List Rooms API Response

Status: **“200/success”**
List of rooms displayed
in JSON format

```
Response 200 / success
{
  "items": [
    {
      "id": "Y2lzY29zcGFyazovL3VzL1JPT00vZDRmYWE3ZWItNmI0Ni0zN2VmLWEzN",
      "title": "Sparky",
      "type": "direct",
      "isLocked": false,
      "lastActivity": "2018-03-26T12:16:37.344Z",
      "creatorId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS9kMzB1ZDBjOC1jOTI1LT",
      "created": "2018-03-26T12:16:10.608Z"
    },
    {
      "id": "Y2lzY29zcGFyazovL3VzL1JPT00vMDI4MmU0YjAtMmMwMS0xMmU4L1TkW",
      "title": "My Postman-Room",
      "type": "group",
      "isLocked": false,
      "lastActivity": "2018-03-26T11:42:28.642Z",
      "creatorId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS9kMzB1ZDBjOC1jOTI1LT",
      "created": "2018-03-21T12:11:39.515Z"
    },
    {
      "id": "Y2lzY29zcGFyazovL3VzL1JPT00vNmM2NGQ1NjAtMmNmMi0xMmU4L1WJkN",
      "title": "API-Team",
      "type": "group",
      "isLocked": true,
      "lastActivity": "2018-03-21T10:26:47.862Z",
      "teamId": "Y2lzY29zcGFyazovL3VzL1RFQU0vNmM2NGQ1NjAtMmNmMi0xMmU4L",
      "creatorId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS9kMzB1ZDBjOC1jOTI1LT",
      "created": "2018-03-21T10:26:47.862Z"
    },
    {
      "id": "Y2lzY29zcGFyazovL3VzL1JPT00vNjU0YjM5NTEtZWY4NC0zYTFjLWFlM",
      "title": "Sheyda Tomkins",
      "type": "direct",
      "isLocked": false,
      "lastActivity": "2018-03-21T09:00:39.001Z",
      "creatorId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS9kMzB1ZDBjOC1jOTI1LT",
      "created": "2018-03-21T09:00:39.001Z"
    }
  ]
}
```


API-Programming / Cisco-Spark / APIC-EM with Python

List Messages Response

Status: **“200 / success”**

List of messages for the room
“Lunch group” in JSON format.

```
Response 200 / success
{
  "items": [
    {
      "id": "Y2lzY29zcGFyazovL3VzL01FU1NBR0UvYzJmOWEwMjAtMzBlYS0xMmU4LTg",
      "roomId": "Y2lzY29zcGFyazovL3VzL1JPT00vMDI4MmU0YjAtMmQwMS0xMmU4LTk",
      "roomType": "group",
      "text": "h8hhohihh",
      "personId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS9kMzBlZDBjOC1jOTI1LTRmN",
      "personEmail": "uwe.starke@hs-wismar.de",
      "html": "h8hhohihh",
      "created": "2018-03-26T11:42:28.642Z"
    },
    {
      "id": "Y2lzY29zcGFyazovL3VzL01FU1NBR0UvYWYyNGY3NzAtMzBlYS0xMmU4LTg",
      "roomId": "Y2lzY29zcGFyazovL3VzL1JPT00vMDI4MmU0YjAtMmQwMS0xMmU4LTk",
      "roomType": "group",
      "text": "iiiiii",
      "personId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS9kMzBlZDBjOC1jOTI1LTRmN",
      "personEmail": "uwe.starke@hs-wismar.de",
      "created": "2018-03-26T11:41:46.983Z"
    },
    {
      "id": "Y2lzY29zcGFyazovL3VzL01FU1NBR0UvNDJkNGNlMjAtMmQwMS0xMmU4LTg",
      "roomId": "Y2lzY29zcGFyazovL3VzL1JPT00vMDI4MmU0YjAtMmQwMS0xMmU4LTk",
      "roomType": "group",
      "text": "Hi all in the room!",
      "personId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS9kMzBlZDBjOC1jOTI1LTRmN",
      "personEmail": "uwe.starke@hs-wismar.de",
      "created": "2018-03-21T12:13:27.426Z"
    }
  ]
}
```



Experimenting with REST APIs using Cisco Spark

Using Postman with the Cisco Spark API

Different Applications – Same Response

Cisco Spark for Developers

Postman

```
Response 200 / success
{
  "items": [
    {
      "id": "Y2lzY29zcGFyazovL3VzL1JPT00vMzRmMWRlYjAtYzU4MS0xMmU3LTg2NzQ0tNG0xMzc0OGI1YTlh",
      "title": "Help Desk",
      "type": "group",
      "isLocked": false,
      "lastActivity": "2017-11-09T19:09:06.155Z",
      "creatorId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS8wODEwNjU4Ny1mOTcwLTQ4Nm00dDl0C1jMDI2NjdjYjE1Mzg",
      "created": "2017-11-09T19:09:06.155Z"
    },
    {
      "id": "Y2lzY29zcGFyazovL3VzL1JPT00vMjhjNmVkdA1tYzU4MS0xMmU3LTg2NzQ0t0TE4M03NTAzMjM5",
      "title": "Network Automation Team",
      "type": "group",
      "isLocked": false,
      "lastActivity": "2017-11-09T19:08:45.741Z",
      "creatorId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS8wODEwNjU4Ny1mOTcwLTQ4Nm00dDl0C1jMDI2NjdjYjE1Mzg",
      "created": "2017-11-09T19:08:45.741Z"
    },
    {
      "id": "Y2lzY29zcGFyazovL3VzL1JPT00vNWU0Mjg2MzI1tYzI5MS0zNDhLLWl1xODYtYjUzZTB1ZTMwNTNh",
      "title": "Sheyda Tomkins",
      "type": "direct",
      "isLocked": false,
      "lastActivity": "2017-11-09T18:33:21.430Z",
      "creatorId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS8wODEwNjU4Ny1mOTcwLTQ4Nm00dDl0C1jMDI2NjdjYjE1Mzg",
      "created": "2017-11-09T18:33:21.430Z"
    }
  ]
}
```

```
Body Cookies Headers (12) Test Results Status: 200 OK Time: 700 ms Size: 2.89 KB
Pretty Raw Preview JSON
1 - {
2 -   "items": [
3 -     {
4 -       "id": "Y2lzY29zcGFyazovL3VzL1JPT00vYmZlZWRhYjAtYzU4YS0xMmU3LTgzYWM0ODk1M2QzODk4ZTdi",
5 -       "title": "My new room",
6 -       "type": "group",
7 -       "isLocked": false,
8 -       "lastActivity": "2017-11-09T20:15:37.435Z",
9 -       "creatorId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS8wODEwNjU4Ny1mOTcwLTQ4Nm00dDl0C1jMDI2NjdjYjE1Mzg",
10 -      "created": "2017-11-09T20:15:37.435Z"
11 -     },
12 -   ],
13 -   {
14 -     "id": "Y2lzY29zcGFyazovL3VzL1JPT00vMjc0ZWl1wMDAtYzU4NS0xMmU3LWVhZjM0ODM1NDhkMTY1ZTU4",
15 -     "title": "Technical Training Classes",
16 -     "type": "group",
17 -     "isLocked": false,
18 -     "lastActivity": "2017-11-09T19:35:33.888Z",
```



Experimenting with REST APIs using Cisco Spark

About Python



- Python's **requests** module provides an easy to use HTTP library to send and receive messages to web servers.
 - Requests is an Apache2 licensed HTTP library
- The Cisco Spark API uses JSON to encode messages.
- To work with JSON encoded data, in Python import the **json** module.
 - This library is used to work with JSON encoded data in Python.
- The **time** module includes timers and functions for convert time formats.

```
import requests
import json
import time
```

21.04.2018



Experimenting with REST APIs using Cisco Spark

Example Using Functions



```
def setHeaders():
    spark_header = {"Authorization": accessToken,
                    "Content-Type": "application/json"}
    return spark_header

def getRooms(theHeader):
    uri = "https://api.ciscospark.com/v1/rooms"
    resp = requests.get(uri, headers=theHeader)
    return resp.json()

header = setHeaders()
value = getRooms(header)
```



API-Programming / Cisco-Spark / APIC-EM with Python



Remember...

You are not expected to completely understand the Python code used in these programs.

Read the comments, edit existing code to meet your needs, and experiment.

```
90
91 while True:
92
93     # Input the name of the room to be searched
94     roomNameToSearch = input("\nWhich room are you looking for? (Can use partial name of the room.) ")
95
96     # Defines a variable that will hold the roomId
97     roomIdToMessage = None
98
99     rooms = r.json()['items']
100 for room in rooms:
101     # Searches for the room 'title' using the variable roomNameToSearch
102     if(room['title'].find(roomNameToSearch) != -1):
103
104         # Displays the rooms found using the variable roomNameToSearch (additional options included)
105         print("\nFound rooms with the word " + roomNameToSearch)
106         print(room['title'])
107         # print ("Room name: " + room['title'])
108         # print ("Room name: " + room['title'] + " ID: " + room['id'])
109
110     # Stores room id and room title into variables
111     roomIdToMessage = room['id']
112     roomTitleToMessage = room['title']
113     break
```





Network Programmability with Cisco APIC-EM



Programming the APIC-EM REST API



Easy to use:

- In mobile apps
- In console apps
- In web apps



Cisco APIC-EM REST APIs

- Hosts
- Devices
- Users
- + more

How does this work?

```

apic-em-examples — bash — 90x20
Hosts=
192.168.68.130
26.6.6.9
26.6.6.11
26.6.6.10
25.5.5.56
14.4.4.12
14.4.4.17
14.4.4.11
14.4.4.10
12.2.2.11
12.2.2.10
12.2.2.12

Policies=
419abccf-c2c9-4421-a96a-f3de3981ce5f
100c8e40-fa50-4faF-a099-c0b0436329c0
2a9460ba-7db2-4790-b1b3-8d13166c9e5
fdbccc11-ca01-4d5f-8bdc-636ecbb6f8b3
2c27beea-ddef-4b77-ada6-a378d392bbe3
  
```



„Programmierung des Netzwerkes ohne Kenntnisse der Console, ein Traum für Nicht-Cisco-Admins?“



Kurs- bzw. Workshopinhalte im Überblick?

Network Programmability with Cisco APIC-EM

1.3.3: Parsing JSON with Python

 [1.3.3.1: Demonstration - Google Map Geocoding API Application](#)

 [1.3.3.2: Authenticating a RESTful Request](#)

 [1.3.3.3: Activity - Get Your Google API Key](#)

 [1.3.3.4: Importing Modules](#)

 [1.3.3.5: Create Variables for API Request](#)


 [1.3.3.6: Activity - Test the URL Request](#)


 [1.3.3.7: Print the URL and Check the Status of the JSON Request](#)

 [1.3.3.8: Activity - Test Status and URL Print Commands](#)

 [1.3.3.9: Extract the Formatted Address Value](#)

2.1.2: The APIC-EM

 [2.1.2.1: What is the Cisco APIC-EM?](#)

 [2.1.2.2: APIC-EM DevNet Login](#)

 [2.1.2.3: APIC-EM Home Page](#)

 [2.1.2.4: Activity - APIC-EM Topology Page](#)

2.2: Programming the APIC-EM REST API

2.2.1: REST APIs

 [2.2.1.1: REST API Review](#)

 [2.2.1.2: REST API Operation](#)

 [2.2.1.3: REST Request Elements](#)

 [2.2.1.4: REST Response Elements](#)

 [2.2.1.5: Authenticating with the APIC-EM](#)

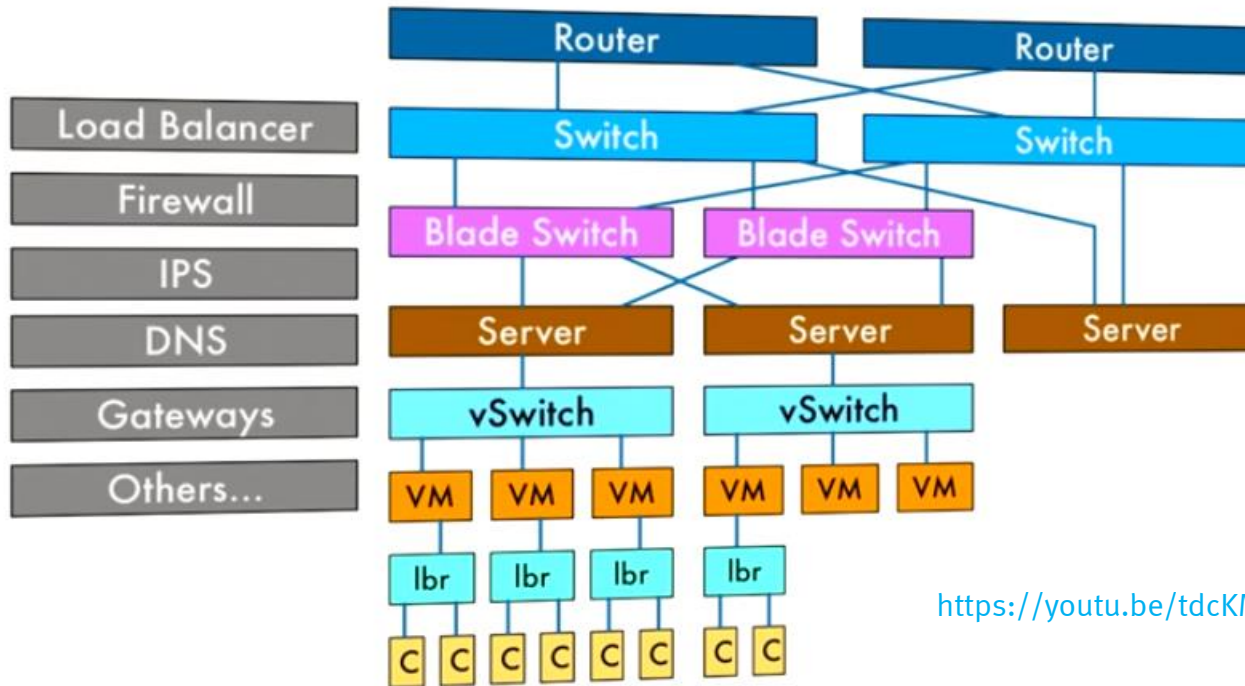


Network Programmability with Cisco APIC-EM

CISCO Networking Academy | DEVNET

Why network programmability“

The Network...



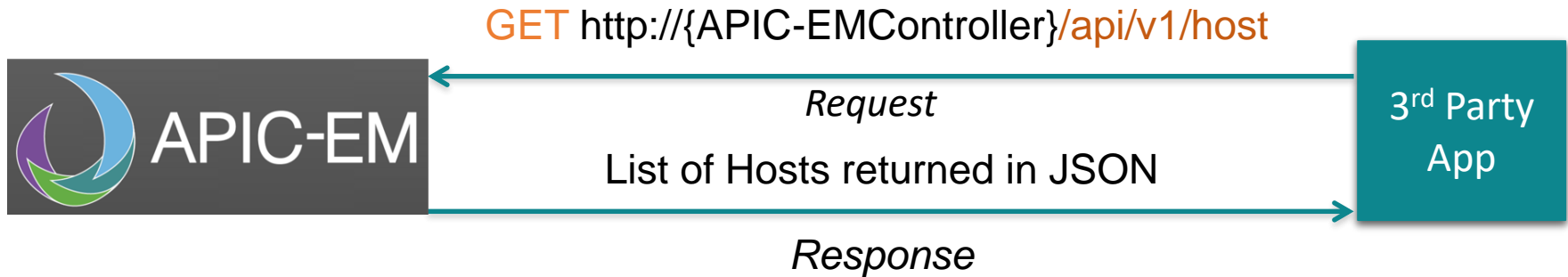
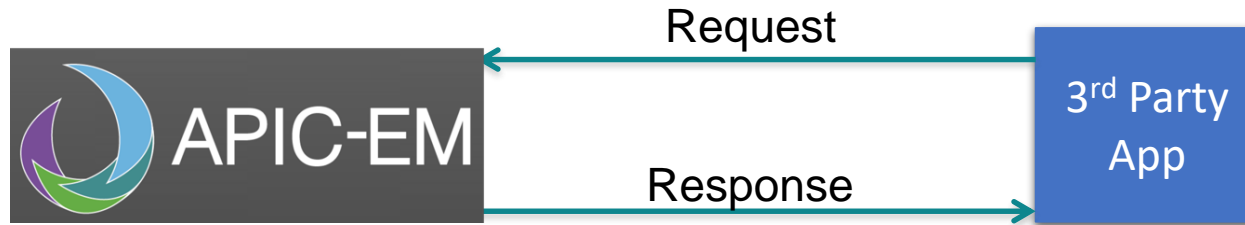
<https://youtu.be/tdcKMJe1UTc>



Network Programmability with Cisco APIC-EM



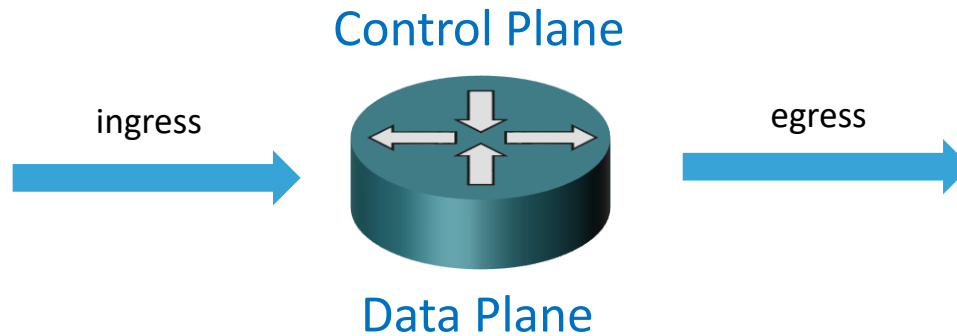
Programming the APIC-EM REST API



SDN: Control Plane and Data Plane

Hardware	Purpose	Example Processes
Device CPU	makes decisions about where traffic is sent	routing protocols, spanning tree, AAA, SNMP, CLI

Regarded as the brains of a device. Used to make forwarding decisions. Information sent to the control plane is processed by the CPU.



Also called the forwarding plane, this plane is the switch fabric connecting the various network ports on a device. The data plane of each device is used to forward traffic flows.

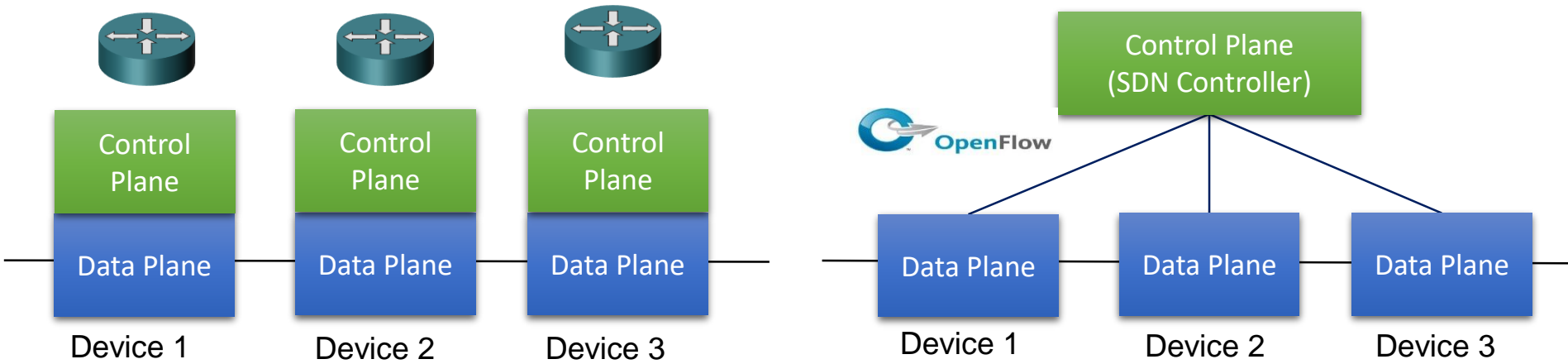
Hardware	Purpose	Example Processes
Dedicated ASICs	forwards traffic to the selected destination	packet switching, L2 switching, MPLS, QOS, policies, ACLs



Network Programmability with Cisco APIC-EM

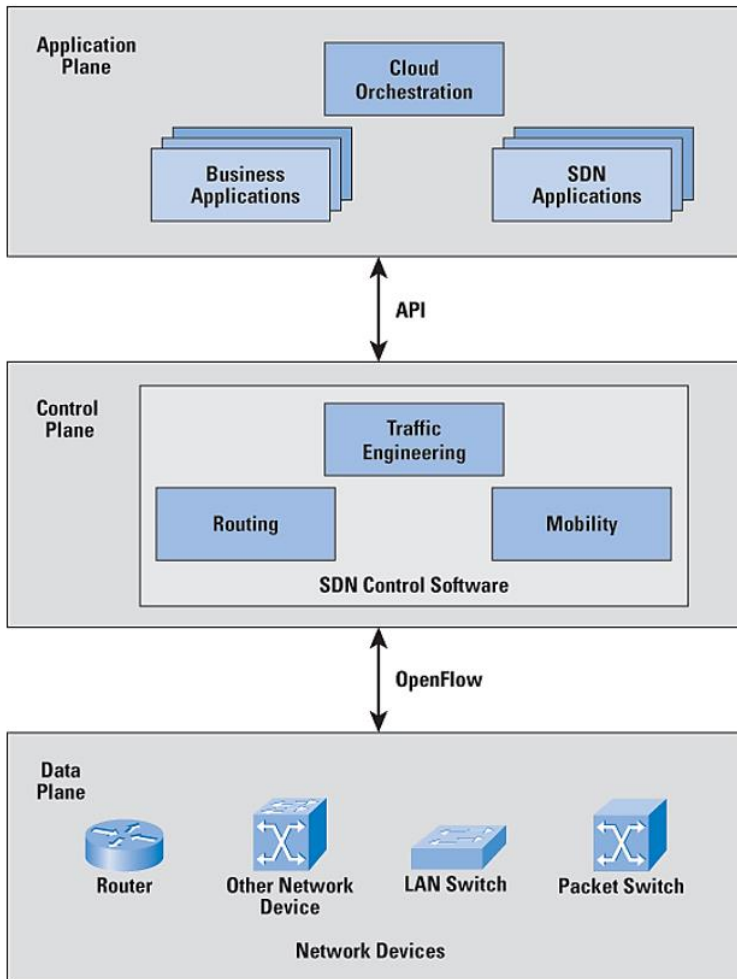


Traditional and SDN Architectures

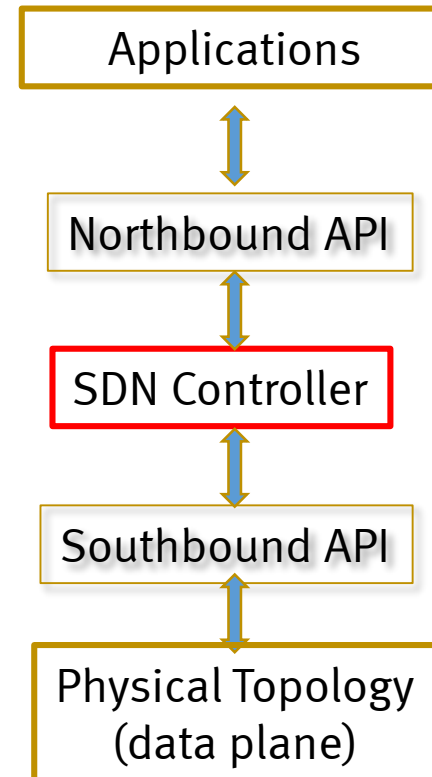




Network Programmability with Cisco APIC-EM



SDN Framework





Network Programmability with Cisco APIC-EM

 | 

What is the APIC-EM?

Cisco Application Policy Infrastructure Controller Enterprise Module (Download for free)

- A Software-Defined Networking (SDN) controller for enterprise networks
- A virtual, software-only, or physical appliance
- Creates an intelligent, open, programmable network with open APIs
- Can transform business-intent policies into dynamic network configuration
- Provides a single point for network-wide automation and control



Not Compatible



Switches

Routers

Compatible



Switches

Routers



Kurs- bzw. Workshopinhalte im Überblick?

Network Programmability with Cisco APIC-EM

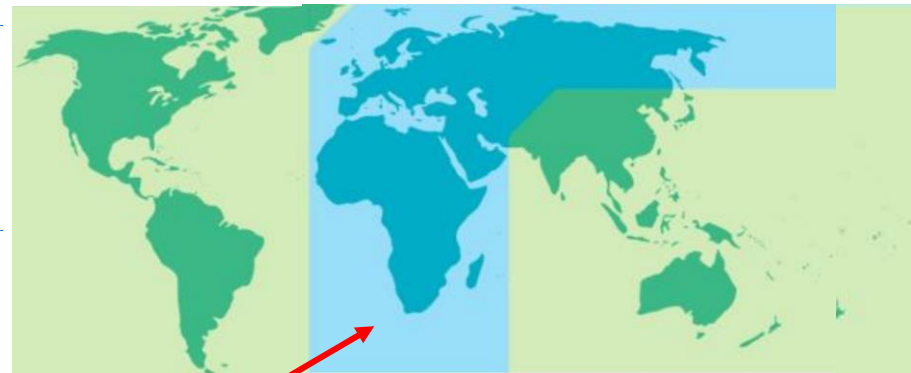
APIC-EM – Log in

Virtualized APIC-EM Controllers are available in several **DevNet Sandboxes:**
Always On, NetAcad instances for NetAcad users only

<https://DevNetSBX-NetAcad-APICEM-1.cisco.com>

<https://DevNetSBX-NetAcad-APICEM-2.cisco.com>

<https://DevNetSBX-NetAcad-APICEM-3.cisco.com>





NOTE: DevNet Scheduled Maintenance

On Thursday April 19 through Sunday April 22, the Cisco DevNet Sandbox will be unavailable due to scheduled maintenance. The maintenance only impacts the Network Programmability with Cisco APIC-EM workshop.

San Francisco:	19 April 1:30 a.m. - 22 April 1:30 a.m. (PDT, UTC-7)
New York:	19 April 4:30 a.m. - 22 April 4:30 a.m. (EDT, UTC-4)
London:	19 April 9:30 a.m. - 22 April 9:30 a.m. (BST, UTC +1)
Hong Kong	19 April 4:30 p.m. - 22 April 4:30 p.m. (HKT, UTC +8)
Sydney:	19 April 6:30 p.m. - 22 April 6:30 p.m. (AEDT, UTC +11)



Network Programmability with Cisco APIC-EM

 Networking Academy | 

How can we use it?

<https://sandboxapicem.cisco.com/>



 CISCO

 **APIC-EM**

Cisco Application Policy Infrastructure Controller Enterprise Module

Username

Password

Log In

User: **devnetuser**
P/W: **Cisco123!**



Network Programmability with Cisco APIC-EM



EPIC-AM Homepage

The screenshot shows the APIC-EM Enterprise Module Home page with the following components:

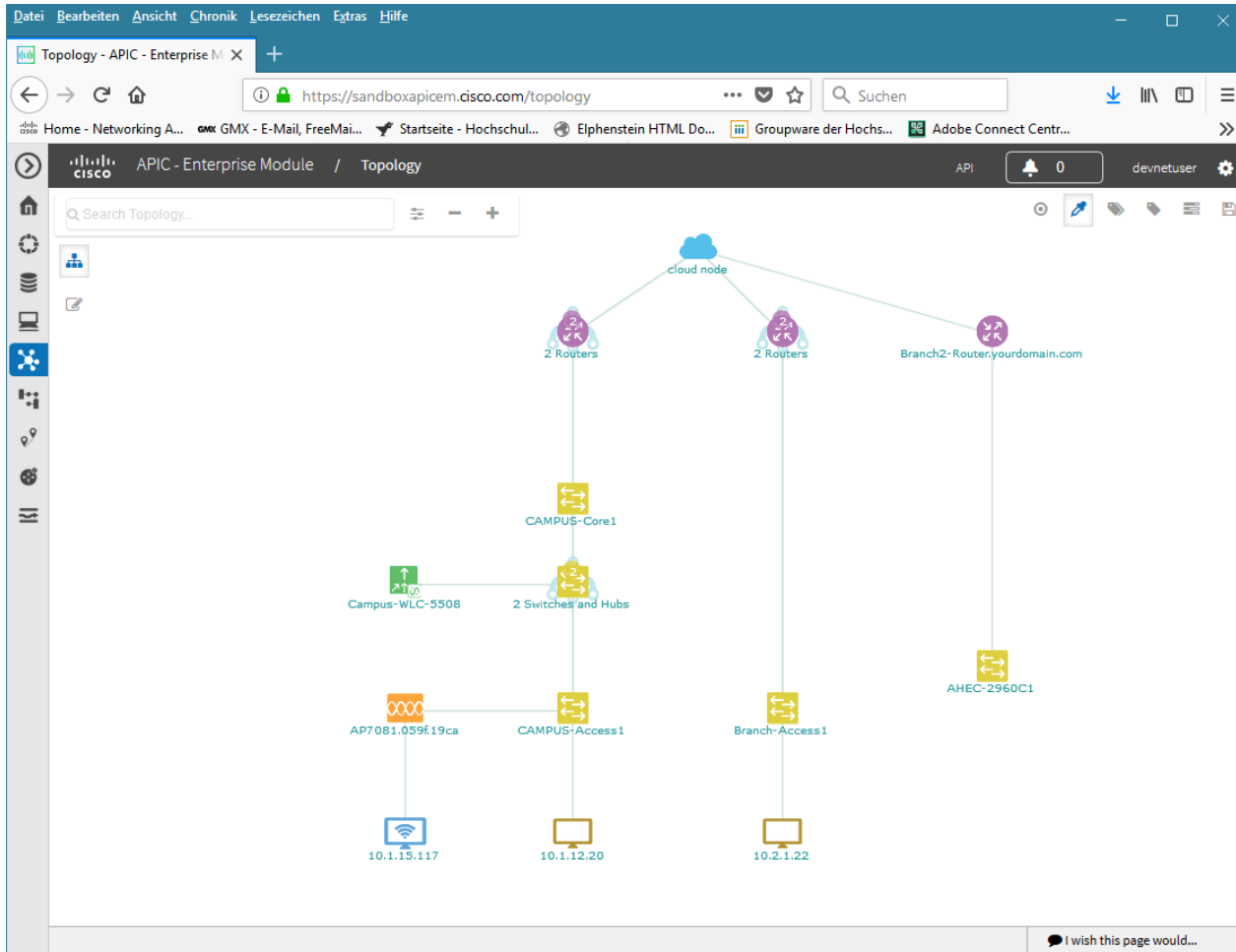
- Navigation:** Home, Discovery, Device Inventory, Host Inventory, Topology, IWAN, Path Trace, Network Plug and Play, EasyQoS.
- Dashboard:**
 - DEVICE INVENTORY:** A donut chart showing 13 Devices total. Legend: Managed (1), In-Progress (12), Collection Failure (0).
 - DISCOVERY - UNREACHABLE DEVICES:** A donut chart showing 2 Unreachable Devices.
 - BRANCH SITES:** A message: "To set up branch sites, use IWAN app!".
 - PATH TRACE:** A donut chart showing 23 Traces total. Legend: Success, Failed.
 - NETWORK PLUG AND PLAY PROJECTS:** A folder icon with 6 projects. Legend: 1 Provisioned, 5 Pre-Provisioned, 0 In-Progress, 0 Failed.
 - EASYQOS SCOPES:** A folder icon with 9 scopes. Legend: 2 Scopes with Devices, 7 Scopes without Devices.



Network Programmability with Cisco APIC-EM



EPIC-AM Topology





Network Programmability with Cisco APIC-EM



Programming the APIC-EM REST API

What about authentication? We get a service ticket via

APIC - Enterprise Module / Swagger

API [0] devnetuser

Available APIs

- File
- Flow Analysis
- Grouping
- IP Geolocation
- IP Pool Manager
- Identity-Manager
- Inventory
- Network Discovery
- Network Plug and Play
- PKI Broker Service
- Policy Administration
- Role Based Access Control**
- Scheduler
- Task
- Topology
- Visibility

Role Based Access Control

APIC-EM Service API based on the Swagger™ 1.2 specification

[Terms of service](#)
[Cisco DevNet](#)

aaa : APIs to register and manage AAA Servers Show/Hide | List Operations | Expand Operations | Raw

role : Role Description API Show/Hide | List Operations | Expand Operations | Raw

ticket : Ticket Management API Show/Hide | List Operations | Expand Operations | Raw

- POST /ticket** addTicket
- POST /ticket/attribute** createTicketAttribute
- GET /ticket/attribute/idletimeout** getIdleTimeout
- GET /ticket/attribute/sessiontimeout** getSessionTimeout
- DELETE /ticket/attribute/{attribute}** deleteTicketAttribute
- DELETE /ticket/{ticket}** deleteTicket

user : User Management API Show/Hide | List Operations | Expand Operations | Raw



Network Programmability with Cisco APIC-EM



Programming the APIC-EM REST API We get a service ticket via Python

```
8 import json          # Import JSON encoder and decoder module
9 import requests      # requests module used to send REST requests to API
10
11 requests.packages.urllib3.disable_warnings() # Disable SSH warnings
12
13 post_url = 'https://sandboxapicem.cisco.com/api/v1/ticket'
14 headers = {'content-type': 'application/json'}
15 body_json = {
16     'username': 'devnetuser',
17     'password': 'Cisco123!'
18 }
19 resp = requests.post(post_url, json.dumps(body_json), headers=headers, verify=False)
20 status = str(resp.status_code)
21 print("Ticket request status: " + status)
22 response_json = resp.json()
23 serviceTicket = response_json['response']['serviceTicket']
24 print("The service ticket number is: " + serviceTicket)
25
```

```
Ticket request status: 200
```

```
The service ticket number is: ST-12210-wN3tffh5MNL7ug4xxUjC-cas
```



Network Programmability with Cisco APIC-EM

 Networking Academy | 

Programming the APIC-EM REST API

What's next, what can we do now?

we will create programs

- these programs will display information about the hosts and devices in the APIC-EM network
- to execute a path trace and show the results

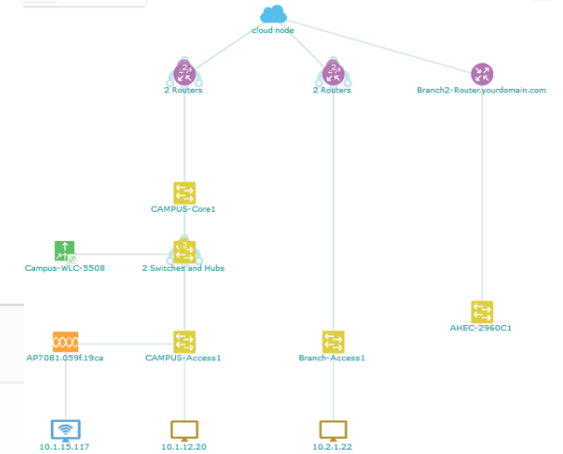
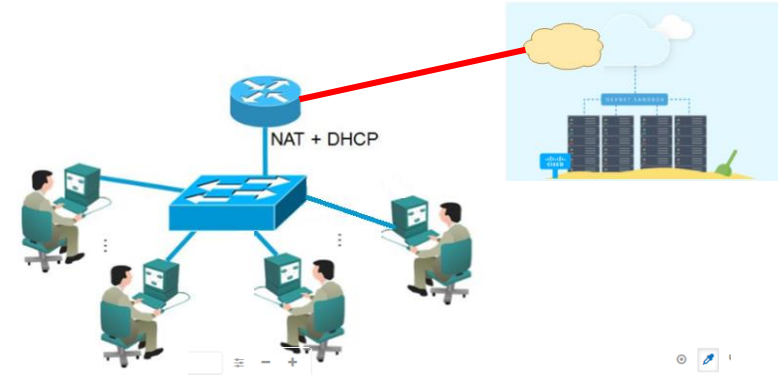


Network Programmability with Cisco APIC-EM



Host inventory results:

```
Status of /host request: 200
=====
Number  Type      IP
=====
      1  wireless 10.1.15.117
      2  wired    10.2.1.22
      3  wired    10.1.12.20
=====
```



Host MAC Address	Host IP Address	Host Type
00:24:d7:43:59:d8	10.1.15.117	WIRELESS
5c:f9:dd:52:07:78	10.2.1.22	WIRED
e8:9a:8f:7a:22:99	10.1.12.20	WIRED



Network Programmability with Cisco APIC-EM



Trace results:

FLOW ANALYSIS ID: c7320ba1-dca5-4520-bd6f-54debd0c52e1
 REQUEST STATUS: COMPLETED
 Path trace:
 Source: 10.2.2.1|
 Destination: 10.2.1.22
 List of devices on path:

Item	Name	IP	Ingress Int	Egress Int
1	Branch-Router1	10.2.2.1	UNKNOWN	GigabitEthernet0/2
2	Branch-Access1	10.2.1.17	GigabitEthernet1/0/2	GigabitEthernet1/0/47
3	Unamed Host	10.2.1.22	UNKNOWN	UNKNOWN

