



BRING YOUR LAPTOP!

Cisco Spark API Workshop

Eugene Morozov
Technical Manager CEE-RCIS, N&B
21 April 2018
Fulda



What is this?

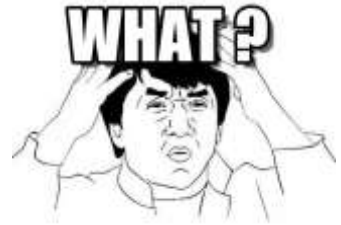
This session IS NOT:


- The upcoming Emerging Technologies Workshop “Experimenting with API...”
- Nor a shortened version of it

This session IS:

- Quick hands-on dive into the topic of API
- Demonstration of power and easiness of use of technology

Cisco Spark → Webex Teams



 **Webex** Products ▾ Plans & Pricing

🔍 Host Join Sign In ▾ [Try it free](#)



Cisco Spark is becoming Cisco Webex Teams.

Webex Teams is an app for continuous teamwork with video meetings, messaging, file sharing and white boarding.

[Download](#)

Get Cisco Spark today and receive an automatic update to Webex Teams as soon as it's ready.



Step 0

- Fill your NetAcad.com email address here:
<http://cs.co/workshop2018>
 - For Spark room
 - For Resources course
- Check your mail for invitation from Spark
- Register and create your Spark account at
<https://www.ciscospark.com/>
- Install Spark mobile or desktop app
- Join the room

Fill your NetAcad.com email address here:
<http://cs.co/workshop2018>

Agenda

- 1 What is API
- 2 Explore Spark API
- 3 Install IDLE and PT7
- 4 Interact with Spark from IDLE
- 5 Interact with Spark from PT7

APIs and RESTful APIs

Application Programming Interface (API)

- An API allows one piece of software talk to another.
- An API is like a power socket.
- Without a power socket, what would you have to do to power your laptop?
 - Open the wall
 - Understand all the wires in the wall
 - Unsheathe wires
 - Splice wires together
- An API defines how a programmer can write a piece of software to extend an existing application's features or even build entirely new applications.



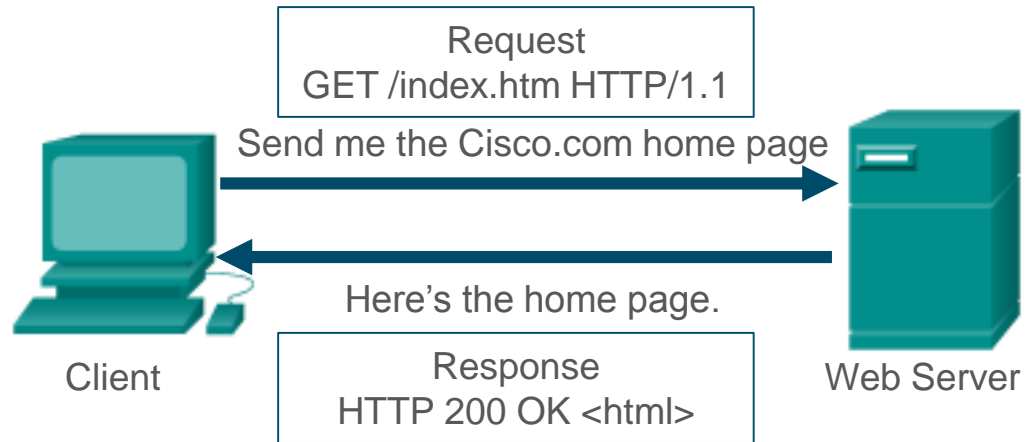
API Example

- **Restaurant Recommendation App**
 - Returns a list of relevant restaurants in the area
 - Integrates a third-party API to provide map functionality
 - The map API enforces a specification of an interface



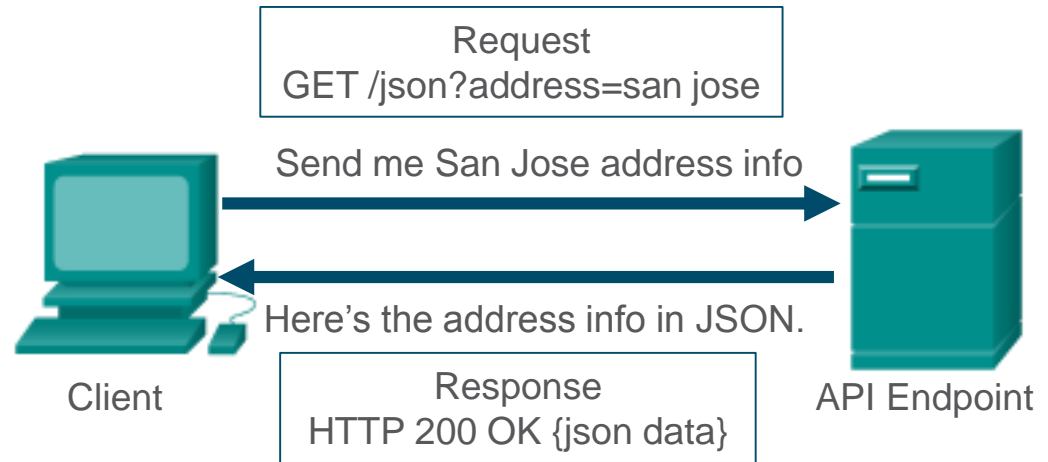
Web Services Interface using HTTP

- Web browsers use Hypertext Transfer Protocol (HTTP) to request (GET) a web page.
- If successfully requested (HTTP status code 200), web servers respond to GET requests with a Hypertext Markup Language (HTML) coded web page.



RESTful API using HTTP

- Representation State Transfer (REST) APIs use HTTP to interface with RESTful services.
- The HTTP request asks for JavaScript Object Notation (JSON) formatted data.
- If successfully formatted according to the API documentation, the server will respond with JSON data.



Anatomy of a RESTful Request

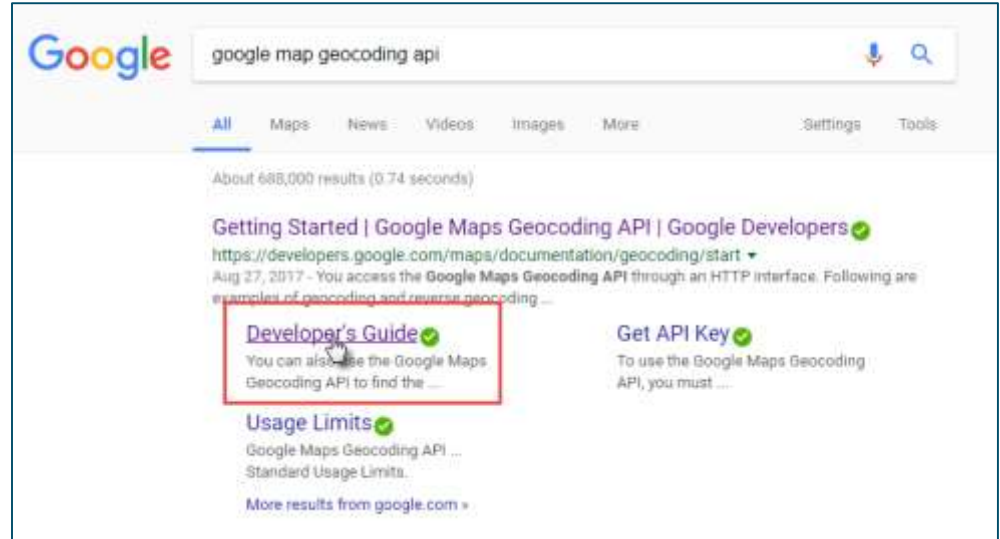
<http://maps.googleapis.com/maps/api/geocode/json?address=sanjose>



- **API Server:** The URL for the server that answers REST requests
- **Resources:** Specifies the API that is being requested.
- **Format:** Usually JSON or XML
- **Parameters:** Specifies what data is being requested

API Documentation

- Use an Internet search to find documentation for an API.



API Documentation

- The API documentation will specify...

- The request **format** (JSON, XML, or text)
- The request **parameters**
- The response format

Google Maps Geocoding API Request Format

A Google Maps Geocoding API request takes the following form:

```
https://maps.googleapis.com/maps/api/geocode/outputFormat?parameters
```

where `outputFormat` may be either of the following values:

- `json` (recommended) indicates output in JavaScript Object Notation (JSON); or
- `xml` indicates output in XML

To access the Google Maps Geocoding API over HTTP, use:

```
http://maps.googleapis.com/maps/api/geocode/outputFormat?parameters
```

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



Explore Spark API

Spark API

- <https://developer.webex.com>
- Docs
- API Reference
- List Rooms
- List Messages
- Test Mode

Tools Installation

Tools Installation

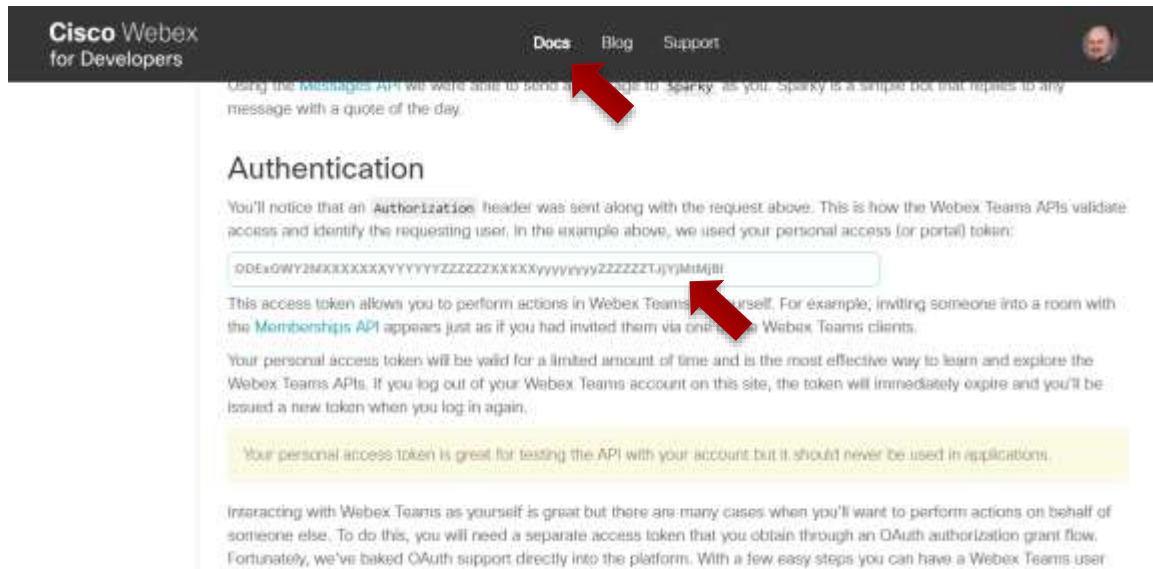
- Python and IDLE
 - Follow steps from the lab scenario Part 1

- Packet Tracer
 - Make sure you have at least PT7.1 installed

Interact with Spark from IDLE

Getting your access token

1. Go to <https://developer.webex.com/>
2. Log in with your Spark account
3. Go to **Docs** page
4. Scroll down the **Getting Started** page down to **Authentication** section
5. Copy your personal **access token**



Cisco Webex
for Developers

Docs Blog Support

Using the [Messages API](#) we were able to send a message to Sparky as you. Sparky is a simple bot that replies to any message with a quote of the day.

Authentication

You'll notice that an **Authorization** header was sent along with the request above. This is how the Webex Teams APIs validate access and identify the requesting user. In the example above, we used your personal access (or portal) token:

```
00Ea09WY2MXlXXXXXXXXYYYYZZZZZZXXXXYYYYYYYYZZZZZZTjYjMhMjBj
```

This access token allows you to perform actions in Webex Teams as yourself. For example, inviting someone into a room with the [Memberships API](#) appears just as if you had invited them via one of the Webex Teams clients.

Your personal access token will be valid for a limited amount of time and is the most effective way to learn and explore the Webex Teams APIs. If you log out of your Webex Teams account on this site, the token will immediately expire and you'll be issued a new token when you log in again.

Your personal access token is great for testing the API with your account but it should never be used in applications.

Interacting with Webex Teams as yourself is great but there are many cases when you'll want to perform actions on behalf of someone else. To do this, you will need a separate access token that you obtain through an OAuth authorization grant flow. Fortunately, we've baked OAuth support directly into the platform. With a few easy steps you can have a Webex Teams user

Getting the room id

1. From the **Docs** page,
2. Under **API Reference**, navigate to **Rooms – List Rooms**
3. Enable **Test Mode**
4. Click **Run**
5. Copy room **id** from the server Response

Cisco Webex for Developers Docs Blog Support

WEBEX TEAMS API

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

List Rooms

List rooms.

The `title` of the room for 1-to-1 rooms will be the display name of the other person.

By default, lists rooms to which the authenticated user belongs.

Long result sets will be split into [pages](#).

GET `https://api.ciscospark.com/v1/rooms`

Test Mode

Request Headers

Content-type	application/json; charset=utf-8
Authorization	<u>Bearer ODExOWY2MDY1YjdmZS00f</u>

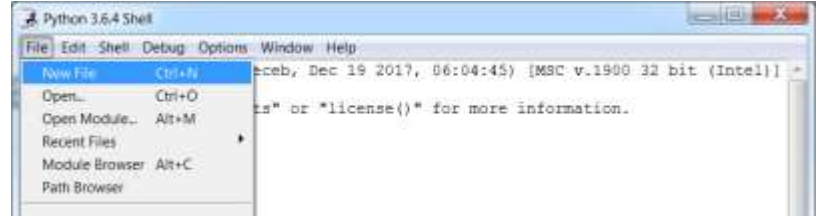
Response 200 / success

```
{
  "items": [
    {
      "id": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "title": "Akademietag Fulda",
      "type": "group",
      "isLocked": false,
      "lastActivity": "2018-04-21T06:41",
      "teamId": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "creatorId": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "created": "2018-04-11T07:33:23.1"
    },
    {
      "id": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "title": "Mugha Balgaonkar",
      "type": "direct",
      "isLocked": false,
      "lastActivity": "2018-04-05T08:32",
      "creatorId": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "created": "2018-04-05T08:32:53.4"
    },
    {
      "id": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "title": "Kostiantyn Herasyenko",
      "type": "direct",
      "isLocked": false,
      "lastActivity": "2018-04-04T13:11",
      "creatorId": "V21vV29zcGFyaz0vL3VzL1E3PF8"
    }
  ]
}
```

Getting messages from the room

8. Start IDLE and create a new file

9. Use this code sample:



```
import json
import requests
```

```
accessToken = "[YOUR ACCESS TOKEN]"
roomId = "[YOUR ROOM ID]"
```

```
headers = {"Authorization": "Bearer " + accessToken,
           "Content-Type": "application/json; charset=utf-8"}
```

```
resp = requests.get("https://api.ciscospark.com/v1/messages?roomId=" + roomId + "&max=5",
                   headers=headers, verify=False)
```

```
status = resp.status_code
print("Request status: " + str(status)+"\n")
```

```
response_json = resp.json()
```

```
print("Last messages in the room: " + json.dumps(response_json, indent=4))
```

Import necessary Python modules

Specify you access token and room id

This performs the HTTP GET request

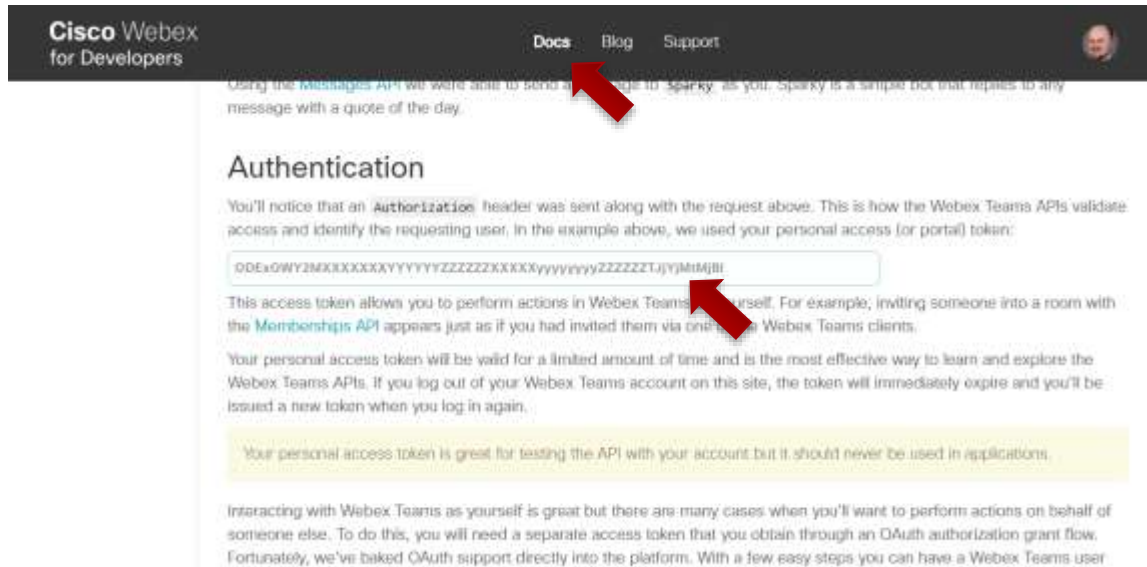
Check if response code OK (200)

Output JSON response

Interact with Spark from PT smart device

Getting your access token

1. Go to <https://developer.webex.com/>
2. Log in with your Spark account
3. Go to **Docs** page
4. Scroll down the **Getting Started** page down to **Authentication** section
5. Copy your personal **access token**



Cisco Webex
for Developers

Docs Blog Support

Using the [Messages API](#) we were able to send a message to Sparky as you. Sparky is a simple bot that replies to any message with a quote of the day.

Authentication

You'll notice that an **Authorization** header was sent along with the request above. This is how the Webex Teams APIs validate access and identify the requesting user. In the example above, we used your personal access (or portal) token:

```
00Ea09WY2MXlXXXXXXXXYYYYZZZZZZXXXXYYYYYYYYZZZZZZTjYjMlMjll
```

This access token allows you to perform actions in Webex Teams as yourself. For example, inviting someone into a room with the [Memberships API](#) appears just as if you had invited them via one of the Webex Teams clients.

Your personal access token will be valid for a limited amount of time and is the most effective way to learn and explore the Webex Teams APIs. If you log out of your Webex Teams account on this site, the token will immediately expire and you'll be issued a new token when you log in again.

Your personal access token is great for testing the API with your account but it should never be used in applications.

Interacting with Webex Teams as yourself is great but there are many cases when you'll want to perform actions on behalf of someone else. To do this, you will need a separate access token that you obtain through an OAuth authorization grant flow. Fortunately, we've baked OAuth support directly into the platform. With a few easy steps you can have a Webex Teams user

Getting the room id

1. From the **Docs** page,
2. Under **API Reference**, navigate to **Rooms – List Rooms**
3. Enable **Test Mode**
4. Click **Run**
5. Copy room **id** from the server Response

Cisco Webex for Developers Docs Blog Support

WEBEX TEAMS API

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

List Rooms

List rooms.

The `title` of the room for 1-to-1 rooms will be the display name of the other person.

By default, lists rooms to which the authenticated user belongs.

Long result sets will be split into [pages](#).

GET `https://api.ciscospark.com/v1/rooms`

Test Mode

Request Headers

Content-type	application/json; charset=utf-8
Authorization	<u>Bearer ODExOWY2MDY1YjdmZS00f</u>

Response 200 / success

```
{
  "items": [
    {
      "id": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "title": "Akademietag Fulda",
      "type": "group",
      "isLocked": false,
      "lastActivity": "2018-04-21T06:41",
      "teamId": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "creatorId": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "created": "2018-04-11T07:33:23.1"
    },
    {
      "id": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "title": "Mugha Balgaonkar",
      "type": "direct",
      "isLocked": false,
      "lastActivity": "2018-04-05T08:32",
      "creatorId": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "created": "2018-04-05T08:32:53.4"
    },
    {
      "id": "V21vV29zcGFyaz0vL3VzL1E3PF8",
      "title": "Kostiantyn Herasyenko",
      "type": "direct",
      "isLocked": false,
      "lastActivity": "2018-04-04T13:11",
      "creatorId": "V21vV29zcGFyaz0vL3VzL1E3PF8"
    }
  ]
}
```


“Real” API

- In PT7, your smart device can communicate with real world using TCP, UDP, and HTTP protocols. Functions that help to do that described in Python API (PT7 → Help → Contents):

Shape Tests Multiuser IPC Internet of Things Using Things Creating Things JavaScript API Python API Visual API	Real HTTP (External Network Access)	Package = realhttp		
	Function	Return Type	Description	Example
	RealHTTPClient()	RealHTTPClient	Creates a Real HTTP Client.	http = RealHTTPClient()
	get(url)	N/A	Gets an URL.	http.get(“http://www.cisco.com”)

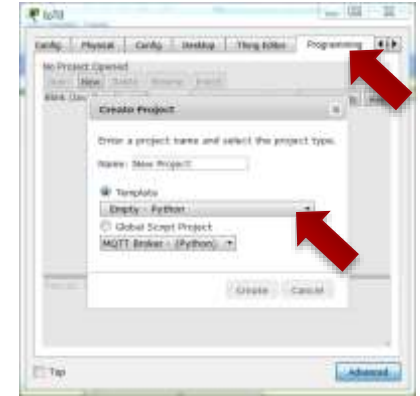
- By default, external network communication is disabled. Enable it in Options → Preferences → Miscellaneous → External Network Access

External Network Access

Enable External Network Access from Device Scripts

Coding for Spark messaging

8. Create new device
9. Go to the **Programming** tab
10. Create a new empty Python project
11. Use this code sample:



```
from realhttp import *
```

```
roomId = "Y21zY29zcZzZzzzYOURroomIdzzzZzZzzzZZZ"  
accessToken = "YmRkYzZzZzzzZzZzzzZzZzzz..."  
message = "1 Test message to Spark from Packet Tracer 7.1!"
```

```
http = RealHTTPClient()
```

```
http.postWithHeader(  
    "https://api.ciscopark.com/v1/messages",  
    {"roomId":roomId,"text":message},  
    {"Authorization":"Bearer "+accessToken, "Content-Type":"application/json"}  
)
```

```
delay(1000)
```

Import additional Python module

Define variables to store room ID, access token and message you want to send

This performs the HTTP POST request

A small delay to let request to be processed

Challenges

Challenge 1:

Pick any smart device and modify its Python code to **send notification to Spark** when device state changes



Challenge 2:

Pick any smart device and modify its Python code to **react on control commands** from the Spark room





Global IPD Week May 7-11

• Program Updates

- Catch up on the latest strategies and products from Cisco Networking Academy!

• Technical Session Topics

- Understanding SNMPv3
- Multilayer Switching
- Cybersecurity - requirements, challenges and growing demand for Security-professionals
- Using Real-World APIs in Packet Tracer
- Behind the Scenes: Creating Netacad Curriculum and Assessments
- Best Practices in Teaching IT Essentials
- Cybersecurity Essentials course Deep Dive

• German Sessions

- Programm Updates
- Eine Technische Sitzung – YOUR session!

Join us for sessions on 7 -11 May 2018.

Click below to register for live sessions, review recordings and download resources. Click the **Archive** button below to see the sessions from previous GIPD Weeks.

The screenshot displays two main categories of sessions: English Sessions and Localized Languages. Under English Sessions, there are two buttons: 'Program Updates 8 May [Check the Agenda]' and 'Technical Sessions 9-10 May [Check the Agenda]'. The Localized Languages section features a grid of buttons for various languages, each with a flag icon and the language name in its native script or Latin characters. The languages listed are: العربية (Arabic), 中文 (Chinese), Русский (Russian), Español (Spanish), Français (French), Italiano (Italian), Türkçe (Turkish), Português (Portuguese), Sinhalese, Hindi, Telugu, Gujarati, Bangla (Bengali), Bahasa (Indonesian), Deutsch (German), Українська (Ukrainian), Polska (Polish), and Hebrew.

Session registration and recordings -
<http://cs.co/GIPD18>