

Automating Mainframe Operations Powering Self-Service Portals with Zowe CLI & Python

Rose Sakach – Offering Manager, Broadcom Open Mainframe
Diego Rodríguez Bravo – Technical Consultant

Disclaimer

Certain information in this presentation may outline Broadcom's general product direction. This presentation shall not serve to (i) affect the rights and/or obligations of Broadcom or its licensees under any existing or future license agreement or services agreement relating to any Broadcom software product; or (ii) amend any product documentation or specifications for any Broadcom software product. This presentation is based on current information and resource allocations as of February 23, 2026, and is **subject to change or withdrawal by Broadcom at any time without notice. The development, release and timing of any features or functionality described in this presentation remain at Broadcom's sole discretion.**

Notwithstanding anything in this presentation to the contrary, upon the general availability of any future Broadcom product release referenced in this presentation, Broadcom may make such release available to new licensees in the form of a regularly scheduled major product release. Such release may be made available to licensees of the product who are active subscribers to Broadcom maintenance and support, on a when and if-available basis. The information in this presentation is not deemed to be incorporated into any contract.

Broadcom may use any feedback provided by you related to a Broadcom product or this presentation for any Broadcom business purposes (including but not limited to, preparation, reproduction, and distribution of derivative works based upon such feedback), without any obligation to you including consent or payment.

Copyright © 2026 Broadcom. All rights reserved. The term "Broadcom" refers to Broadcom Inc. and/or its subsidiaries. Broadcom, the pulse logo, Connecting everything, CA Technologies and the CA Technologies logo are among the trademarks of Broadcom.

THIS PRESENTATION IS FOR YOUR INFORMATIONAL PURPOSES ONLY. Broadcom assumes no responsibility for the accuracy or completeness of the information. TO THE EXTENT PERMITTED BY APPLICABLE LAW, BROADCOM PROVIDES THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. In no event will Broadcom be liable for any loss or damage, direct or indirect, in connection with this presentation, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if Broadcom is expressly advised in advance of the possibility of such damages.

What's Driving Us to Python?

A focus on business results

- ▶ Cost & Performance
- ▶ Compete & Grow
- ▶ Trust & Compliance



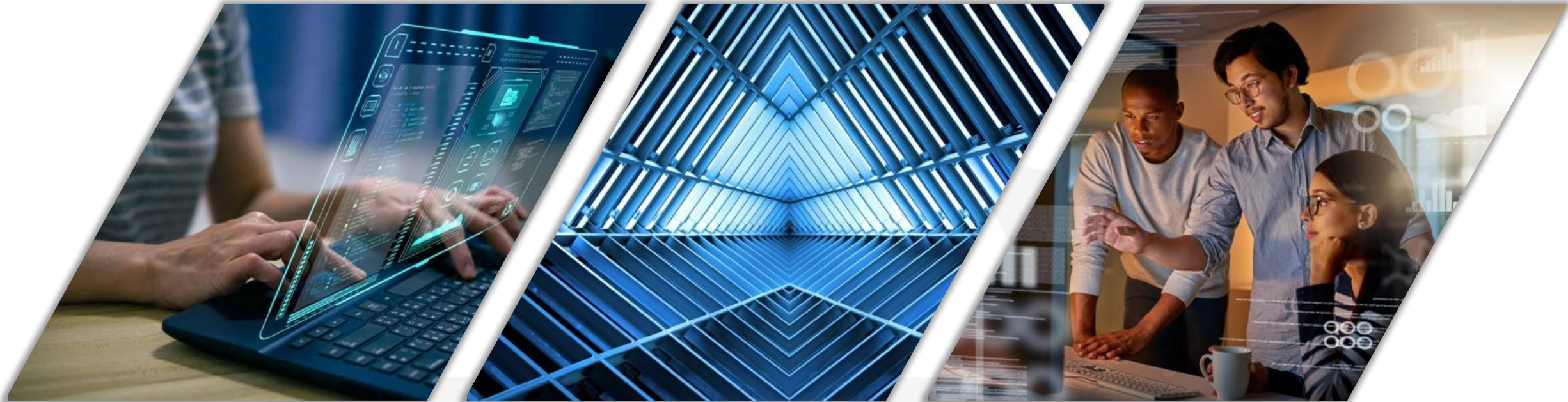
Hands Down, In-Demand Use Cases

Driving the need for interoperability

**SEAMLESS
ONBOARDING**

**AUTOMATE APP
& DATA PIPELINES**

**CROSS-PLATFORM
OBSERVABILITY**



ZERO TRUST

APIs* are at the heart of modernization

Application programming interface

Cross product / cross platform

Communication protocol

Popular industry standard

Recently adopted by mainframe vendors

(Restful) API's are used for data operations

Application programming interface

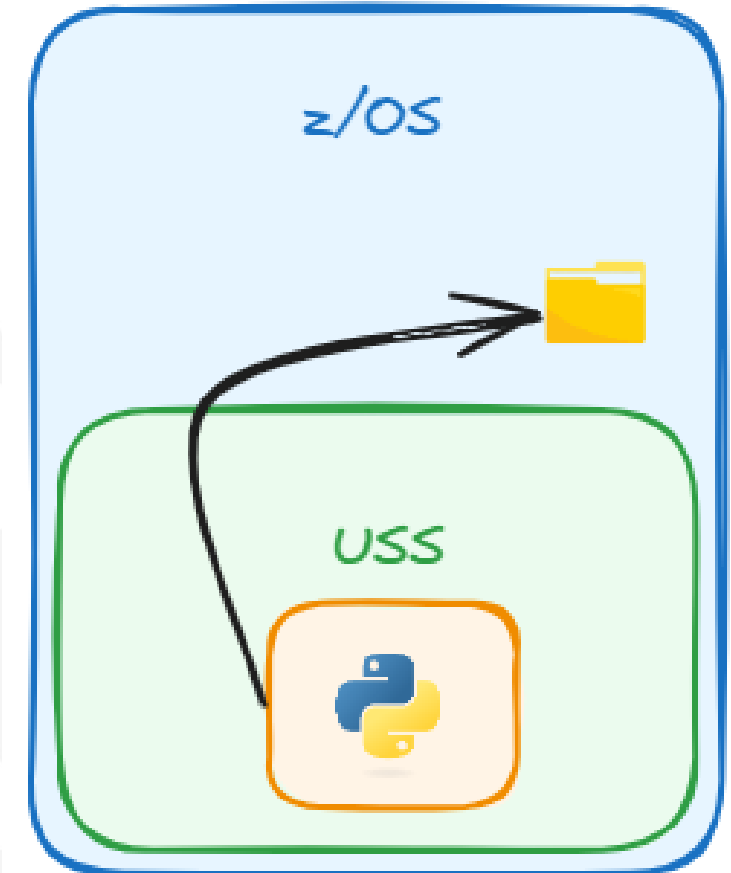
Type of software :



An application programming interface is a way for two or more computer programs or components to communicate with each other. It is a type of software interface, offering a service to other pieces of software. [Wikipedia](#)

Introduction - Python

- **Modern & Extensible:**
 - Integration: APIs, databases, and cloud services
 - Flexible and future-proof automation solution
- **Readability & Maintainability:**
 - Easy to write
 - Debug
 - Maintain
- **Rich Ecosystem:**
 - Vast library of modules
 - Including data analytics, web services, and AI-driven decision-making
- **Cross-Platform Compatibility:**
 - On and Off the mainframe, enabling hybrid IT operations



Running Python in the Mainframe

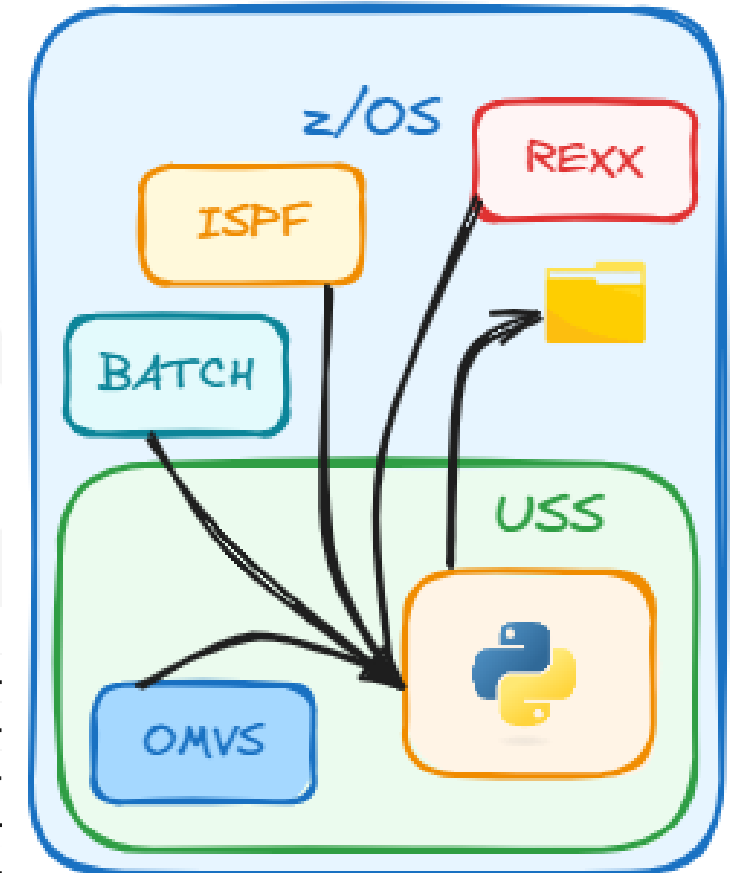
- BATCH

```
//STEP1 EXEC PGM=BPXBATCH
//STDERR DD SYSOUT=*
//STDOUT DD SYSOUT=*
//STDPARM DD *
sh python /u/users/user01/python-mainframe/myScript.py
/*
```

- REXX

```
>>-bpxwunix--(--cmd--+-----
                        '-,stdin--+-----
                        '-,stdout--+-----
                        '-,stderr--+-----
                        '-,env--+-----
                        '-,login--+-----
                        '-,batch-'
```

- OMVS: `python myScript.py`



Python & SSH

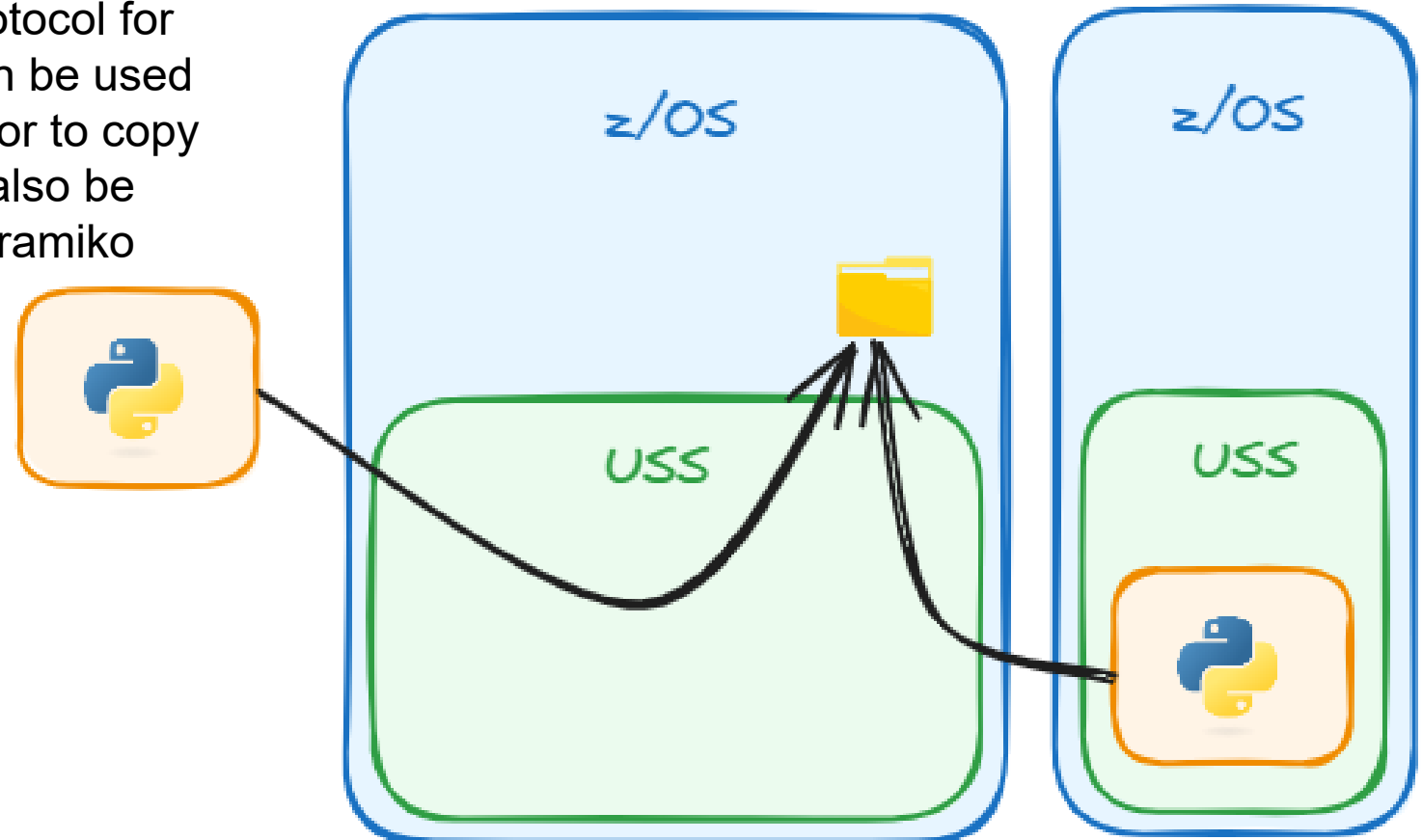
SSH (Secure Shell) is a network protocol for accessing a terminal remotely. It can be used as an interactive shell or pipe (ssh) or to copy items (scp). It is encrypted. It can also be used in Python with libraries like Paramiko

There are 4 common authentication methods, including:

- Username / Password
- Public/Private Key Authentication
- Ticket Based Authentication
- Digital Certificates

Factors to consider:

- Complexity
- Security
- Compliance

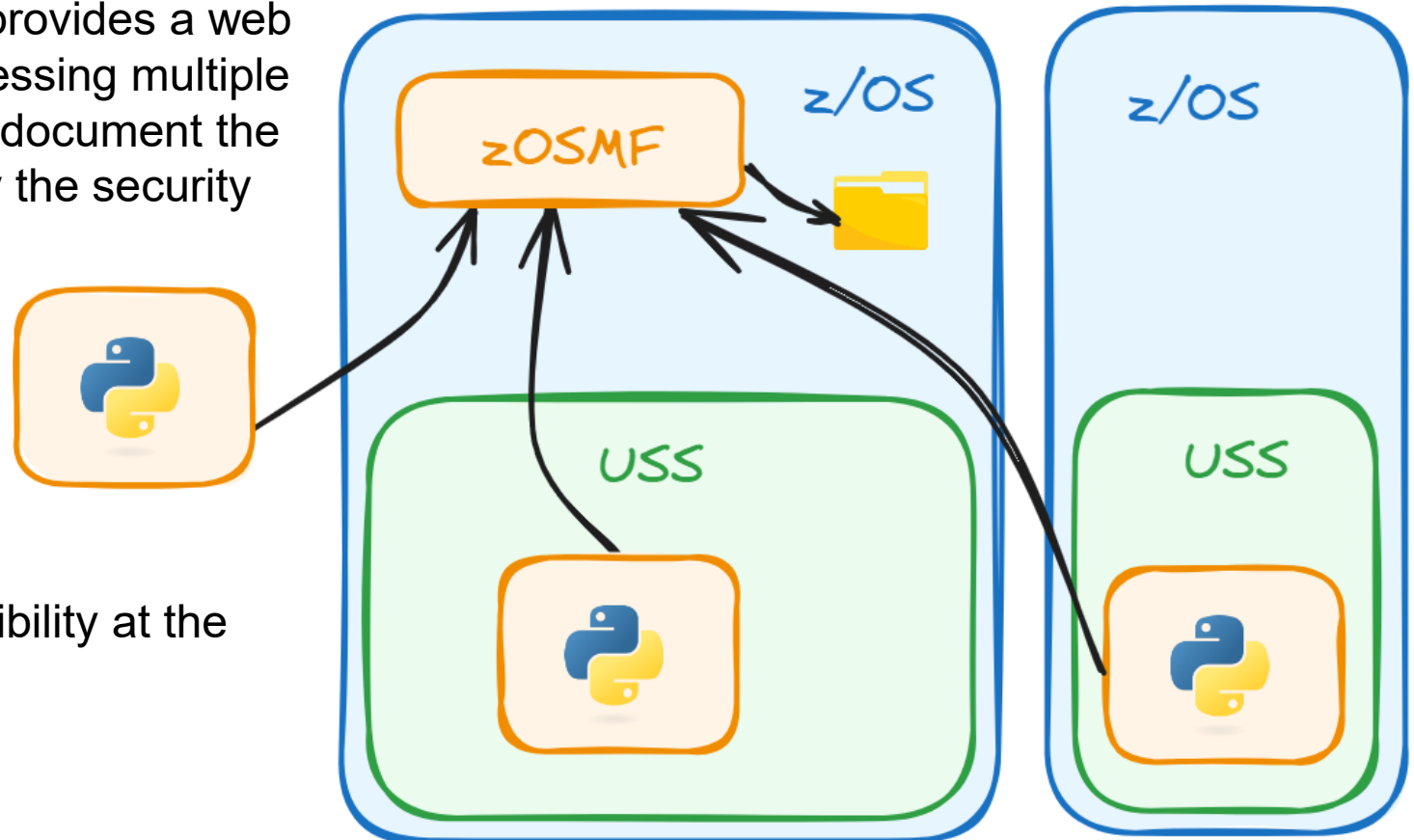


Python & z/OSMF

z/OS Management Facility (z/OSMF) provides a web based GUI and a RESTful API for accessing multiple functions of z/OS. It uses Swagger to document the APIs, but it must also be configured by the security administrator.

Python uses the **requests** library to work with REST APIs.

Using REST, you are working with the lowest level. This gives maximum flexibility at the expense of writing all the wrappers.



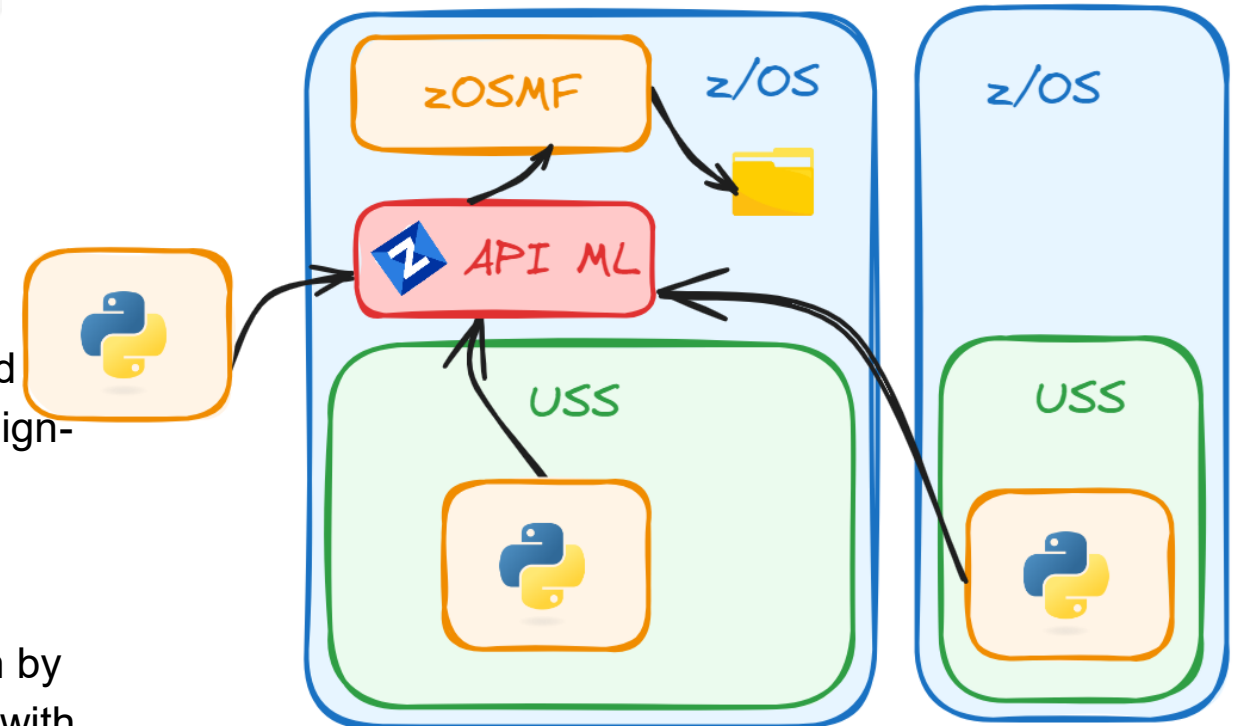
Python & Zowe API ML

Zowe API Mediation Layer

Manages REST APIs, security, unified interface

Main Features:

- API Discovery: Centralized catalog for finding and accessing available APIs on the mainframe.
- Secure Communication: Provides authentication and authorization mechanisms (e.g., support for single sign-on and TLS).
- High Availability: Ensures reliable access to APIs through load balancing and failover support.
- Routing and Integration: Simplifies API consumption by routing requests and enabling seamless integration with other systems.
- User Interface: Offers a web-based interface to browse and explore available APIs.
- Auditing and Logging: Tracks and logs API usage for monitoring and troubleshooting purposes.

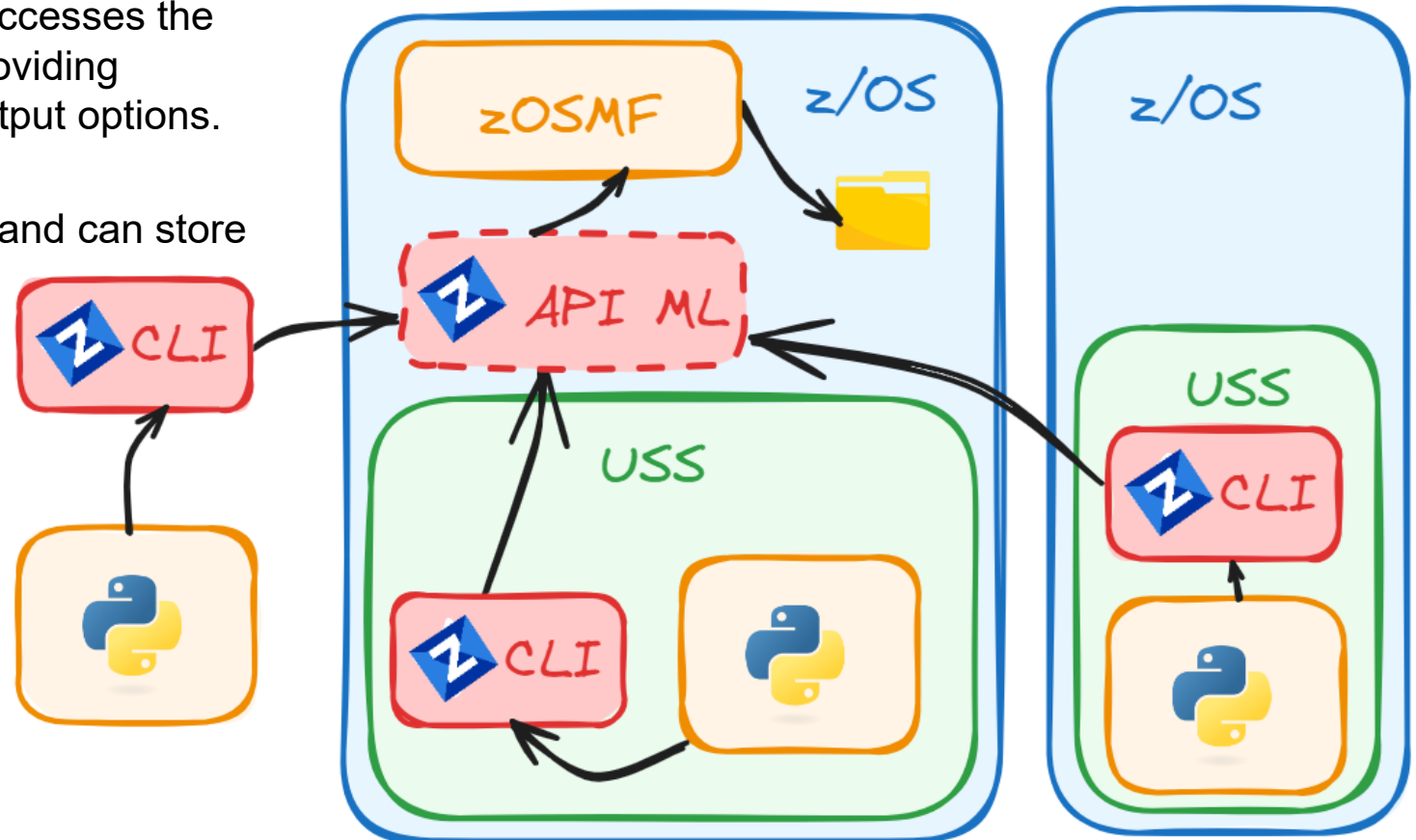


Python & Zowe CLI

The Zowe Command Line Interface (CLI) accesses the REST APIs, however it also wraps them providing additional functionality, such as different output options.

It uses profiles to manage the connections and can store the credentials in an OS based secure credential store. This includes SSO tokens making it easier to manage for highly secured environments.

Files, Jobs, Commands, plugins



Building Web Portals with Python + Zowe CLI

Why Zowe CLI

- Unified access to z/OS resources via REST APIs.
- JSON output – ideal for integration with Python, making it easy to parse return data from Zowe.
- Team Configuration – shared profiles and secure tokens.
- Extensible through plugins and custom commands.
- Bridges DevOps tools (Jenkins, GitHub Actions, VS Code) with mainframe automation.

Why Web

- Lightweight client entirely in a browser
- Familiar experience
- Unified access point – Central hub for mainframe tasks, dashboards, and apps
- Instant updates
- Enterprise-grade security
- Cross-platform – Windows, macOS, Linux, tablets and phones
- Modern user experience – Responsive, visual, and consistent with enterprise web standards.

Create Zowe Utils Reusable Lib (zw.py)

```
import zw  
-----  
import subprocess  
def execute_command(command):  
    try:  
        result = subprocess.run(command, shell=True, text=True, capture_output=True)  
    except subprocess.CalledProcessError as e:  
        print("Command execution failed.")  
        print("Return Code:", e.returncode)  
        print("Error:", e.stderr)  
    if result.returncode!=0:  
        print('rc', result.returncode)  
        print('sto ', result.stdout)  
        print('ste ', result.stderr)  
    return result.stdout, result.stderr, result.returncode
```

Create Zowe Utils Reusable Lib (zw.py) cont.

```
def issue_console_command(command, prof=''):
    command=f'zowe zos-console issue command "{command}" {prof}'
    print(command)
    sto, ste, rc = execute_command(command)
    return(sto,ste,rc)
```

→ `sto,ste,rc=zw.issue_console_command('D SMF')`

```
def cre_trs_file(file):
    dcb = "--r1 1024 --bs 1024 --rf FB --sz 250CYL --ss 250"
    command=f'zowe zos-files create ps "{file}" {dcb}'
    sto, ste, rc = execute_command(command)
    return(sto,ste,rc)
```

→ `sto,ste,rc=zw.cre_trs_file(file_name)`

UI Framework (Streamlit)



Upload & Unterse

The dataset is: prod001.mytest.gs - Elapsed time: 27.55 seconds

Select zowe profile:

Data Set Type: SEQ PDS

Pop in VSCode

File upload

Drag and drop file here
Limit 200MB per file

FILE.Q11.TRS 32.0KB

Name on target

Widgets

```
if st.button("Submit"):
    ...

st.selectbox('Check zOSMF profile',
            temp_zsopf_list,
            key='sb_check_zosmf_value',
            on_change=reset_other_selectboxes,
            args=('sb_check_zosmf_value',))
```

```
pop = st.toggle("Pop in VSCode")
```

```
with st.spinner("Refreshing defaults...")
```

Script Python like workflow

```
# create remote terse dataset
progress_text='Creating Terse dataset...'
my_bar.progress(40, text=progress_text)
sto,ste,rc=zw.cre_trs_file(remote_tersed_dataset)
if rc==1:
    error_msg=ste
    raise MyException

# upload local file binary
progress_text='Uploading local file...'
my_bar.progress(50, text=progress_text)
sto,ste,rc=zw.upload_zos_file_bin(local_file, remote_tersed_dataset, '-b')
if rc==1:
    error_msg=ste
    raise MyException
```



Upload & Unterse

Select zowe profile

Choose an option

Data Set Type:

SEQ
 PDS

Pop in VSCode

File upload



Drag and drop file here
Limit 200MB per file

Browse files

Name on target

Text input field for name on target

SUBMIT



Menu on Sidebar ?

Set Credentials

user :
host :
esm :
port :

Execution Mode

- Test
Display Commands.
- Production
Execute Commands.



Welcome

Welcome to the Zowe ESM Self Service Concept Portal.

Set your default connection credentials from the sidebar to use the utilities.

Select the utility you want to use from the top menus.

IPL Map

Show IPL MAP

Choose an option ▼

Create IPL MAP

Choose an option ▼

Zowe CLI Check Team Config

Check z/OSMF Connection



Check JWT Connection



DataFrame:

	zOSMF access	ssh access	zftp access	tso access	JWT access	JWT creation date	JWT expiration date	Create JWT	Logout JWT
b043	✓	🚀	🚀	🚀	n/a	n/a	n/a	n/a	n/a
b043_ssh_privkey	n/a	PROD001	n/a	n/a	n/a	n/a	n/a	n/a	n/a
b043_apiml	n/a	n/a	n/a	n/a	🚀	🚀	🚀	✗	🚀
r126	🚀	🚀	🚀	✓	n/a	n/a	n/a	n/a	n/a
ca31	🚀	🚀	n/a	n/a	n/a	n/a	n/a	n/a	n/a
ca32	🚀	🚀	🚀	n/a	n/a	n/a	n/a	n/a	n/a
ca32_apiml	n/a	n/a	n/a	n/a	🚀	10-15 18:34 UTC	10-16 02:34 UTC	✓	🚀
ca39	🚀	🚀	✗	🚀	n/a	n/a	n/a	n/a	n/a
svw	🚀	🚀	🚀	n/a	n/a	n/a	n/a	n/a	n/a
svw_apiml	n/a	n/a	n/a	n/a	🚀	🚀	🚀	🚀	🚀

Your Path Forward

Practical next steps

Contact:

(Diego)

diego.rodriquez@broadcom.com

(Rose)

rose.sakach@broadcom.com

Investigate

Zowe:

<https://zowe.org>

<https://docs.zowe.org/>

Python:

<https://docs.python.org/>

[3/reference/index.html](https://docs.python.org/3/reference/index.html)

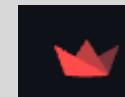
Experiment

Streamlit:

<https://streamlit.io/>

<https://docs.streamlit.io/>

Deploy



Zowe Training Opportunities

HANDS-ON TRAINING



Zowe API Mediation Layer: Intermediate – Digital Badge 06ZWE2001C

Earners of this badge curriculum have a solid understanding of installing, configuring, administering, and troubleshooting Open Mainframe Project Zowe® API Mediation Layer (API ML).



Zowe API Mediation Layer Training 06ZWE20011 4 hrs

Module 1 of a 4-module series offers practical z/OSMF installation tips, basic configuration settings and troubleshooting for Zowe API Mediation Layer. The series includes hands-on administration training, advanced configuration & security facilities with practical troubleshooting examples, challenges and guidance.



Zowe API Mediation Layer: Digital Badge Assessment 06ZWE2000E 1 hr

SELF-PACED TRAINING

Getting Started with the Zowe API Mediation Layer
06ZWE10080 | 30 min

Preparing for the Installation and Configuration of the Zowe API Mediation Layer
06ZWE10050M | 20 min

Configuring the Zowe API Mediation Layer
06ZWE10060 | 30 min

Team Config for Zowe
06ZWE10040M | 45 min

Configuring the Zowe API Mediation Layer for High Availability, Security, and Performance
06ZWE10012M | 40 min

Zowe Explorer - What's New, What's Planned, What's Next
06ZWE100060M | 30 min

CA Spool Plugin for Zowe CLI
06ZWX100060M | 30 min

Technical Education Direct from the Experts

In-Person

European MTE
Prague, Czech Republic
Apr. 21-23, 2026

Virtual

Virtual MTE
Held Virtually
June 23-25, 2026

In-Person

North American MTE
Plano, TX
Oct. 20-22, 2026

Join a Mainframe Technical Exchange

- Network with peers and Mainframe technical experts
- Participate in technical how-to sessions and hands-on workshops, hear product updates, provide feedback
- No registration fee!



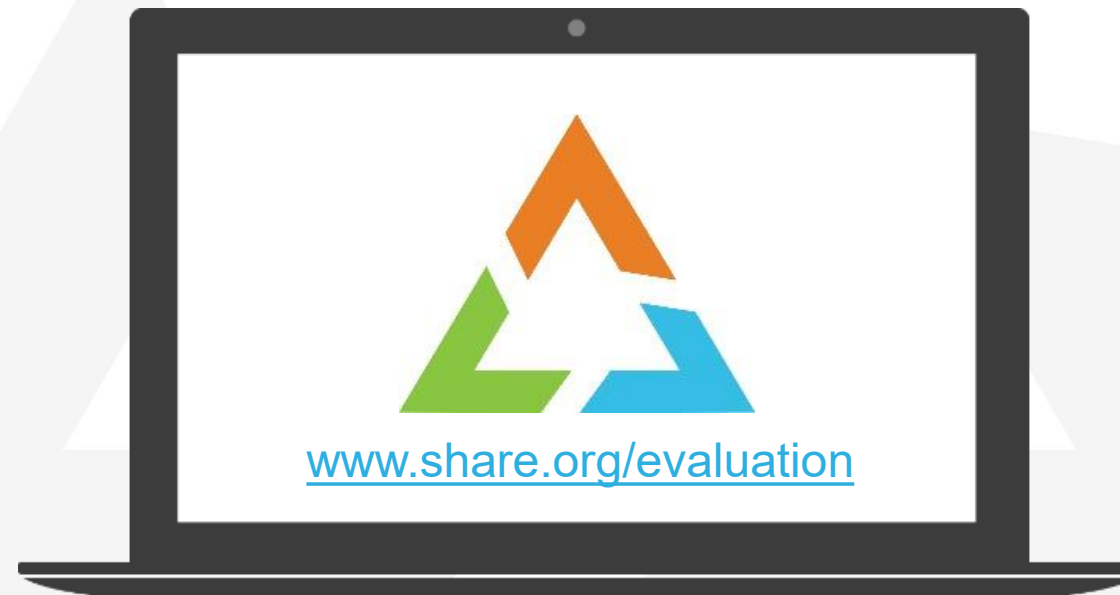
"The event covered all the relevant topics, and the content was amazing. The hands-on experience was also a good way to get actual experience with the tools."

Senthil Mathur, Manager, Accenture

Your feedback is important!

Submit a session evaluation for each session you attend:

www.share.org/evaluation





THANK YOU