



Zowe

Project Overview

zModernization Summit Event

May 7th 2019

Dan Jast
IBM Z Technical Specialist
Poughkeepsie Client Experience Center

Agenda



- Introducing Zowe
- Quick Facts about Zowe
- Why Zowe?
- Zowe Vision Statement
- What's in Zowe?
 - *Zowe REST Services*
 - *Zowe Web Desktop*
 - *Zowe CLI*
 - *Zowe API Mediation Layer*
- Getting Started with Zowe
- "Demo"

Introducing Zowe



- An extensible framework for connecting applications and tools to mainframe data and applications.
- Aims to make the mainframe an integrated and agile platform within the changing IT architectural landscape.
- First open source project on z/OS. All code is licensed under the Eclipse Public License version 2.0



Zowe

Quick Facts about Zowe



- **Zowe 1.0.0 Announce at THINK SF 2019**
 - 100% Open Source (EPL 2.0)
 - Defined extensions points
 - Framework ready for commercial exploitation
- Pronounced as “Zoe” – [zoh-ee] in English
 - Not an acronym – just a simple, fun and easy name
 - Using the spelling “Zowe” allowed us to trademark
- An open source project under the Open Mainframe Project (OMP), a collaborative project within the Linux Foundation
- IBM, Rocket Software and CA Technologies are founding members
- **Generally Available on Feb 8th, 2019**



Zowe



Why Zowe ?



Information Technology is undergoing a revolution of changing architectures



Co-Existence With Other
Cloud Models



Protecting Current and
Future Investments



Simple and Familiar



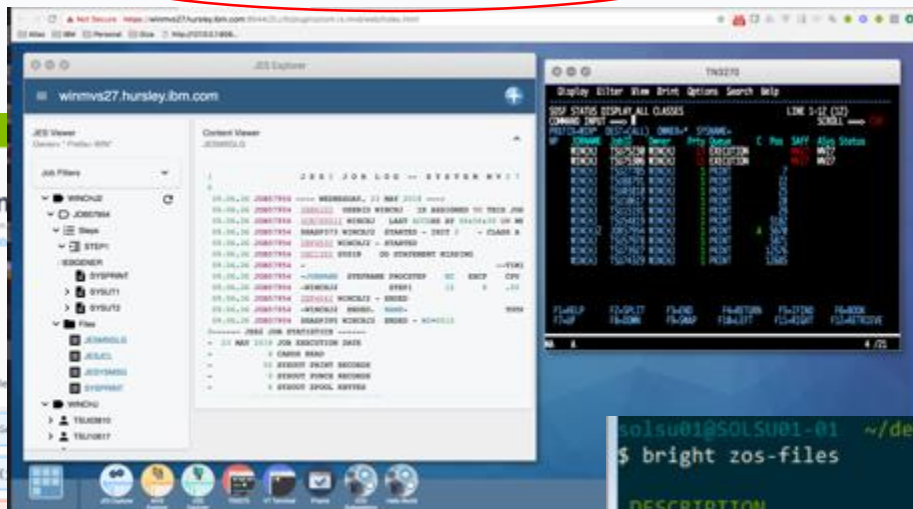
Zowe Vision Statement

- **Attract new people**
 - ✓ Demystify the Z platform
 - ✓ Enhance integration and consumability
 - ✓ Promote Open community of practice
- **Reduce learning curve**
 - ✓ Improve productivity
 - ✓ Modern, platform-neutral interfaces
 - ✓ Cloud-like experience
- **Simplify architecture**
 - ✓ Reduce operational overhead
 - ✓ Improve co-existence
 - ✓ Enable rich ecosystem of free and commercial solutions

What's in Zowe?



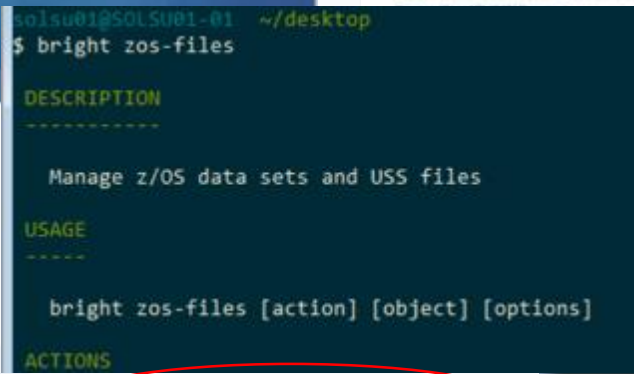
Browser-based Web Desktop



API Mediation Layer (Gateway, Discovery Service, Catalog)



Swagger-defined z/OS REST APIs



Node.js- based CLI

Zowe REST Services – API economy for deep integration



- Industry standard REST interfaces to z/OS resources that are language and platform neutral, stateless and scalable
- Foundational building blocks for system services

- ***Dataset APIs***

- Create, read, update, delete, and list data sets

- ***JES APIs***

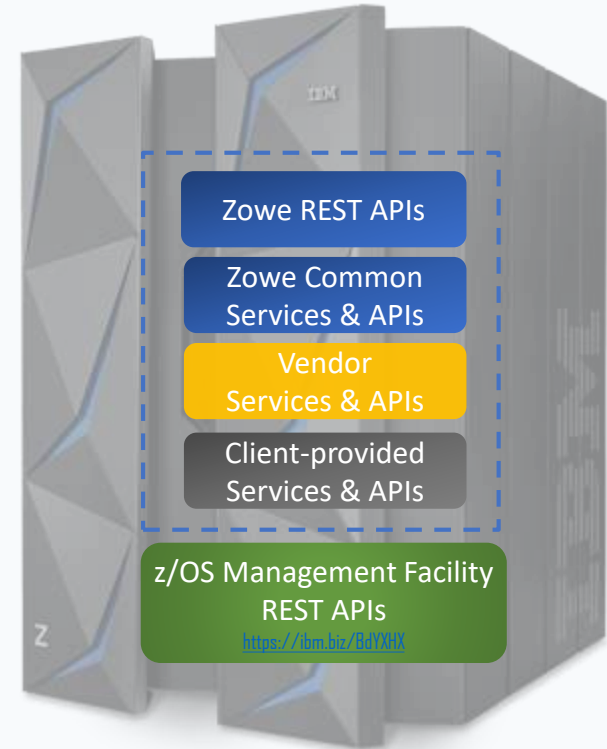
- View the information and files of jobs, and submit and cancel job

- ***USS APIs***

- Create, read, update, and delete USS files

- ***System APIs***

- View information about PARMLIB, SYSPLEX, and USER





- Enables a single point of access to mainframe APIs with high-availability, scalability, dynamic API discovery, consistent security, “one-time” sign-on experience and unified standard API documentation (OpenAPI / Swagger)

- ***API Catalog***

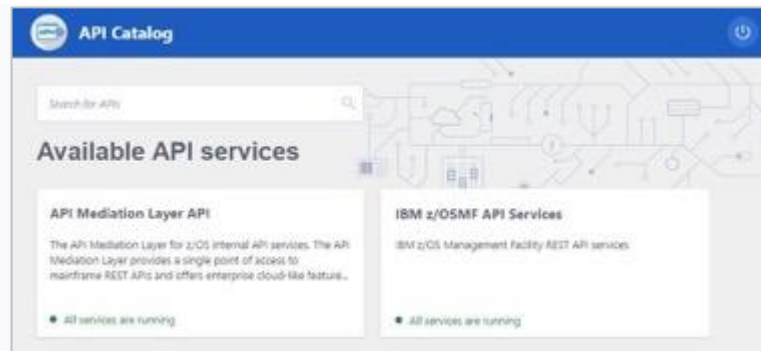
UI Catalog of available APIs with their Swagger doc and service status

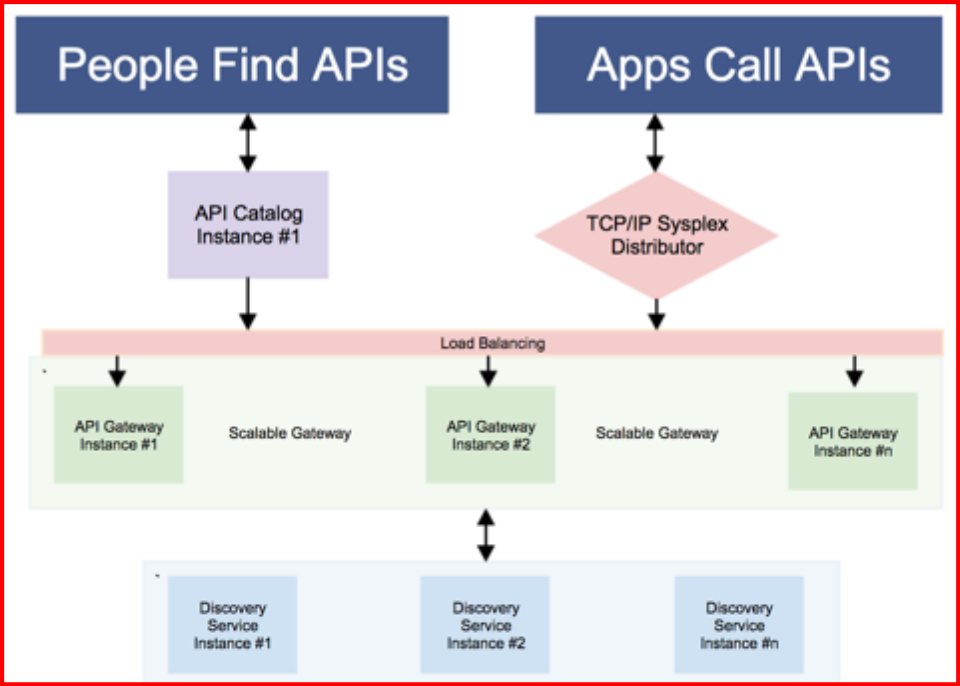
- ***Gateway***

Single secure point of entry to an ecosystem of API services. Hides complexity. Highly available. Based on Netflix Zuul.

- ***Discovery Service***

Discover APIs across many applications. Repository of active API services. Based on Netflix Eureka.

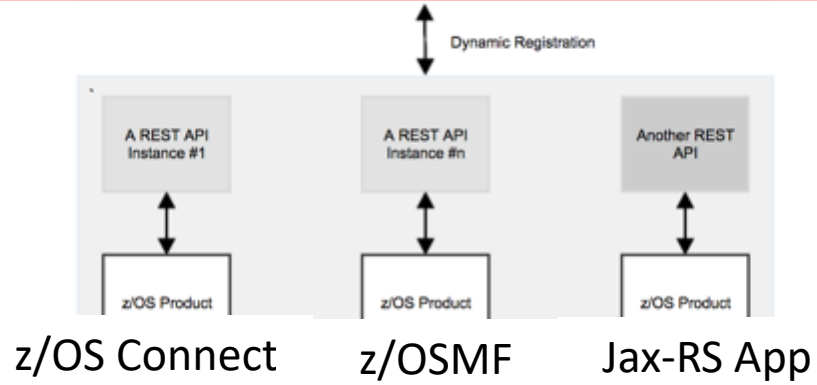




API Layer Components*

- **API Catalog**
UI Catalog of available APIs with their Swagger doc and service status
- **API Gateway**
Single point of entry to an ecosystem of microservices. Hides complexity. Highly available. Based on Netflix Zuul.
- **Discovery Service**
Discover APIs across many applications. Repository of active services. Based on Netflix Eureka.
- **z/OSMF API**
Authenticate Zowe users with mainframe credentials

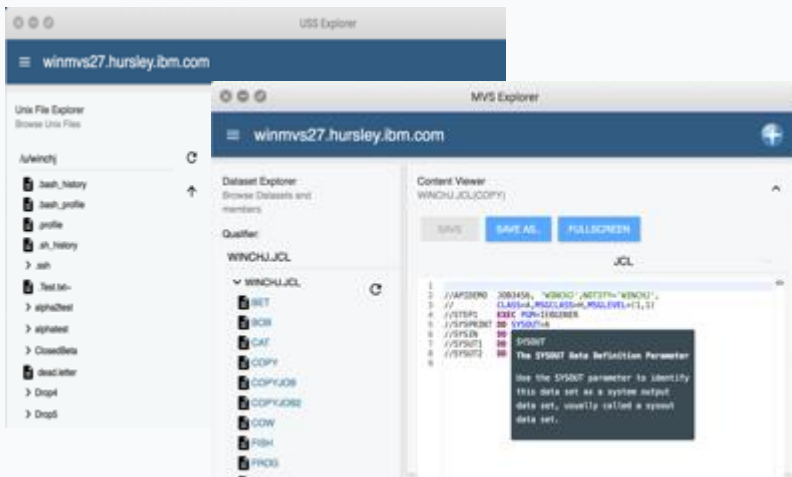
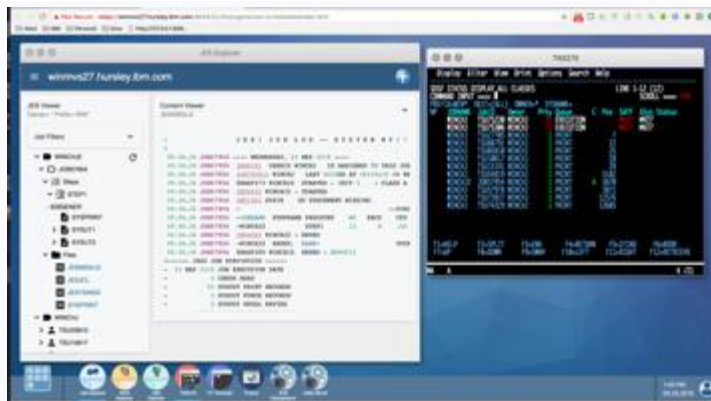
* Separate microservices, might be running as separate address spaces



Zowe Web Desktop – An app container in a browser



- Known as **zLUX**, the Zowe Web UI is a virtual desktop system that offers a rich and open platform for a web-based mainframe user experience.
- **Mainframe Virtual Desktop**
 - A web-based window manager that provides full screen interactive experience
- **Zowe Node Server**
 - Runs zLUX; uses Express.js as web service framework for communication between applications and z/OS services and components, pre-reqs Node.js for z/OS
- **ZSS Server**
 - Provides secured REST API services
- **Application plug-in**
 - Dataservices, Configuration dataservice, URI broker, app-to-app communication, Error reporting UI, Logging utility
- **Explorers**
 - JES, MVS, USS explorers
 - Basic editing support for REXX and JCL



Zowe CLI – Enables cloud-like access to mainframe



- Enables app developer and DevOps engineers to interact with the mainframe easily through a CLI from any terminal on Windows, MacOS, Linux
- Easily integrates with IDEs, shell commands, bash scripts, and build tools; installs using NPM

- ***Interact with mainframe files***

Create, edit, download, and upload mainframe files (data sets) directly

- ***Submit jobs***

Submit JCL from data sets or local storage, monitor status, view and download output automatically

- ***Issue TSO and z/OS console commands***

Issue TSO and console commands to the mainframe directly

- ***Integrate z/OS actions into scripts***

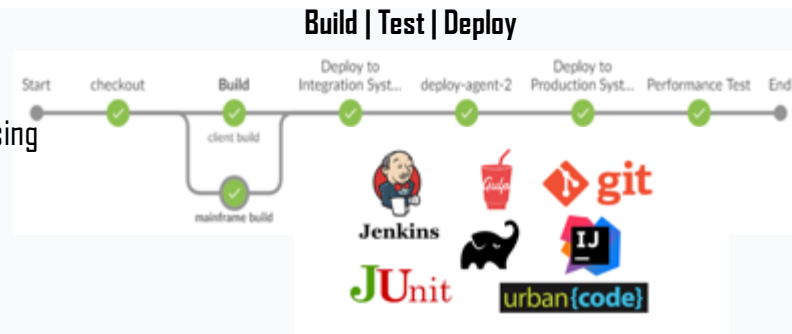
Build local scripts that accomplish both mainframe and local tasks

- ***Produce responses as JSON documents***

Return data in JSON format on request for consumption in other programming languages

- ***CLI Plug-Ins***

Access to CICS and DB2



```
GROUPS
-----

plugins          Install and manage plug-ins
profiles         Create and manage configuration profiles
provisioning | pv Perform z/OSMF provisioning tasks on Published Templates
                  in the Service Catalog and Provisioned Instances in the
                  Service Registry.

zos-console | console Issue z/OS console commands and collect responses
zos-files | files      Manage z/OS data sets
zos-jobs | jobs        Manage z/OS jobs
zos-tso | tso          Issue TSO commands and interact with TSO address spaces
zosmf            Interact with z/OSMF

OPTIONS
-----

--version | -v (boolean)

    Display the current version of CA Brightside

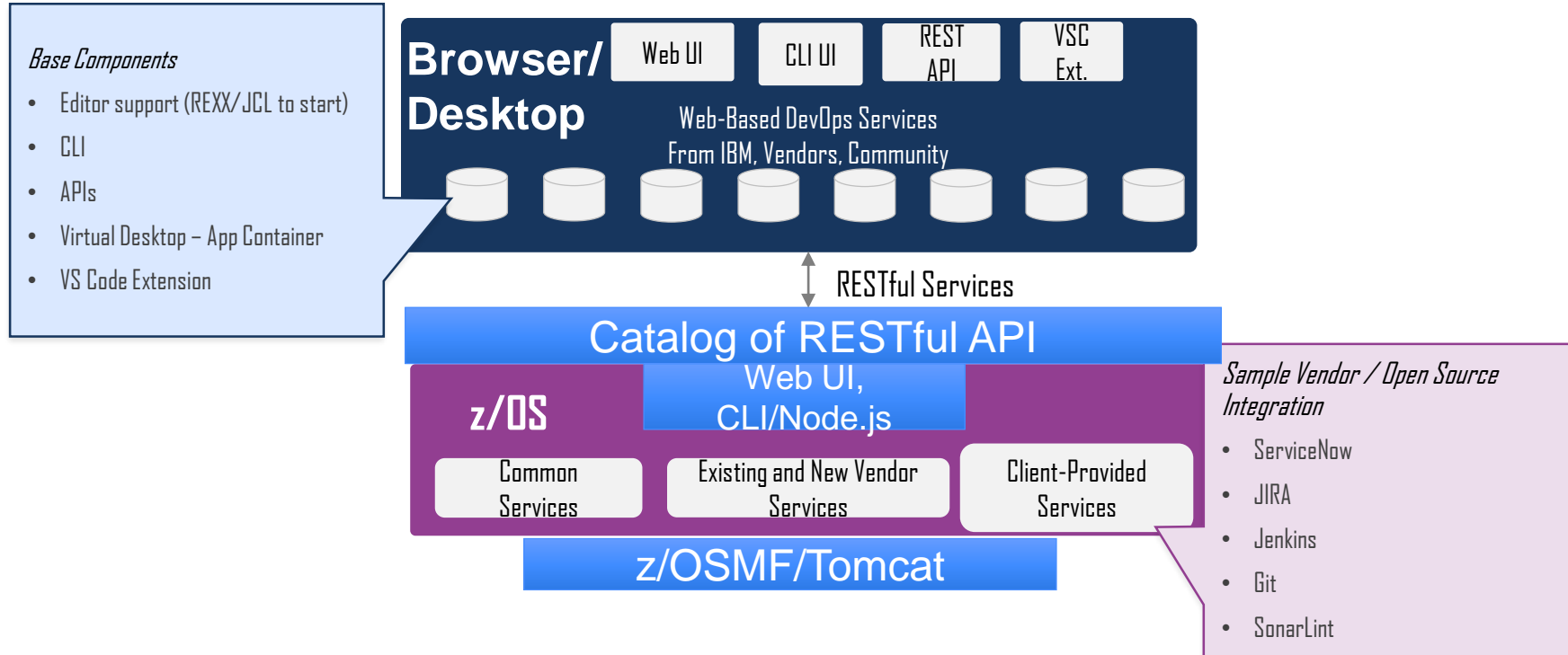
GLOBAL OPTIONS
-----

--response-format-json | --rfj (boolean)

    Produce the command response as a JSON document

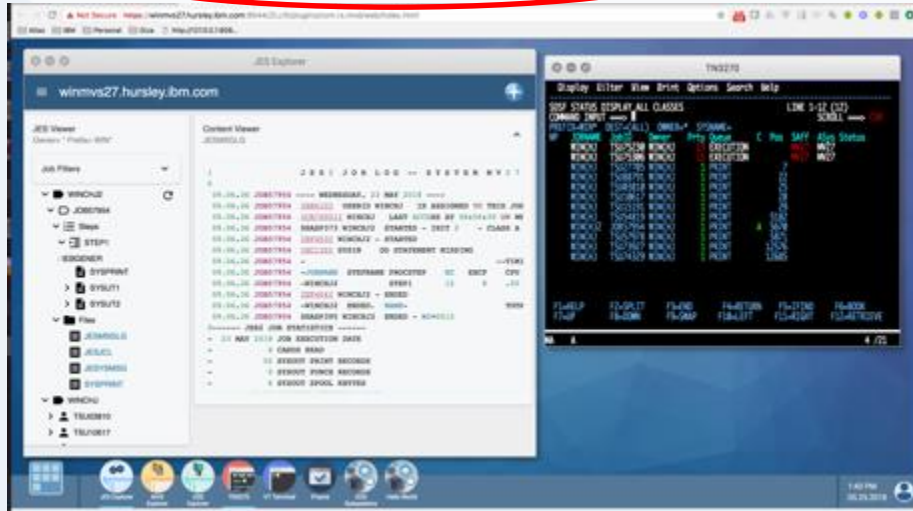
--help | -h (boolean)
```

Zowe High Level Architecture

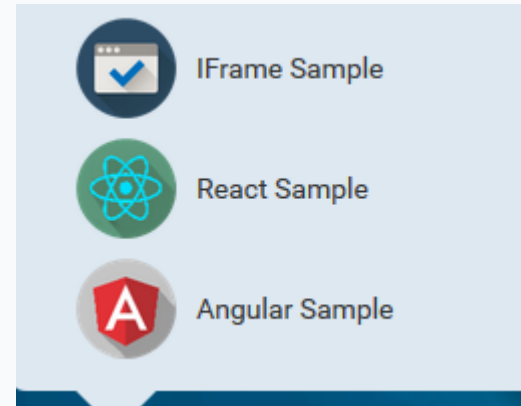




Browser-based Web Desktop



- z/OS Native Web UI for applications
- Launch in context (i.e., right mouse click 3270 to web app)
- App to app communication
- Exploit graphic widgets planned for inclusion



Where is Zowe Extensible?



- REST API enable your products
 - REST API for product controls/admin
 - Sharing of information

A screenshot of a Swagger API documentation page. The page title is "Api Documentation" and it lists several REST APIs for "JES job APIs" under the "Jobs Controller". The APIs are listed in a table with columns for the HTTP method (GET, DELETE, POST) and the endpoint path. The endpoints include /api/v1/jobs, /api/v1/jobs/{jobName}/{jobId}, /api/v1/jobs/{jobName}/{jobId}/files, /api/v1/jobs/{jobName}/{jobId}/files/{fileId}/content, /api/v1/jobs/{jobName}/{jobId}/steps, and /api/v1/jobs/dataset.

- Opt in to API Mediation
- Participate in Single Sign On, High Availability and Status tracking capabilities

API Mediation Layer
(API Catalog, Discovery Service, Gateway)

A screenshot of the "API Mediation Layer API" page. The page title is "API Mediation Layer API" and it provides a description of the layer. Below the description, there is a section for "API Catalog" which includes the API version (1.0.0) and a REST API for the API Catalog service. The page also includes a section for "API Catalog" with a dropdown menu for "Current state information" and a "GET" button for "containers" to list catalog dashboard files. There is also a "GET" button for "containers[id]" to retrieve a specific dashboard file information. The page ends with a section for "API Documentation" with a dropdown menu for "Service documentation".

Swagger-defined z/OS REST APIs

Where is Zowe Extensible?



Node.js- based CLI

```
GROUPS
-----
plugins          Install and manage plug
profiles         Create and manage confi
provisioning | pv Perform z/OSMF provisio
                  in the Service Catalog a
                  Service Registry.
zos-console | console Issue z/OS console comm
zos-files | files   Manage z/OS data sets
zos-jobs | jobs    Manage z/OS jobs
zos-tso | tso      Issue TSO commands and
zosmf             Interact with z/OSMF

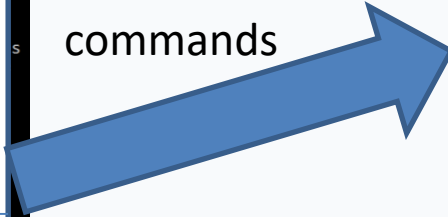
OPTIONS
-----
--version | -v (boolean)
                  Display the current version of CA Brightsi

GLOBAL OPTIONS
-----
--response-format-json | --rfj (boolean)
                  Produce the command response as a JSON document
--help | -h (boolean)
```

- zos-files DS
- zos-files US
- zos-jobs
- TSO
- Console

- “plug-ins”

Out of box commands



z/OSMF

REST APIs
TSO, Console
JES, MVS. USS

Custom Extensions



Your application,
product,
tool, ...

The Zowe open community

Zowe Leadership
Committee (ZLC)

Zowe Continuous
Integration/Continuo
us Development
(CI/CD)

Zowe On-
boarding

Zowe Core
Technology

Zowe API
Mediation
and Security

Mission: Develop the material and supporting activities for onboarding developers and customers

Mission: expand upon the base technologies being contributed to the project

Mission: expand upon integration and interface extension points and overall security

Open
Mainframe
Project Board

Open
Mainframe
Technical
Steering
Committee

Open
Mainframe
Marketing
Committee

Check out

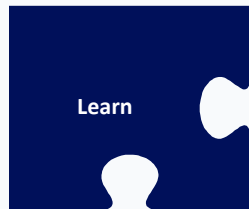
<https://zowe.org/about-us/>

A foundational principle of this new project is meritocracy. The more that somebody contributes, the more responsibility they will earn. A pattern of quality contribution to a project may lead to an invitation to join the project as a committer.

Leadership roles in the Project are also merit-based and earned by peer acclaim. Merit must be demonstrated in publicly-accessible forums. Committers and project leads are added to a project via an election.



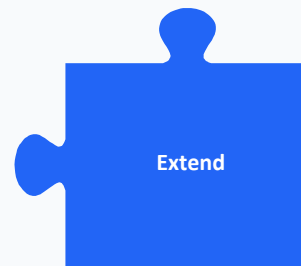
Getting Involved



- Visit the [Open Mainframe Project](#)
- Visit [Zowe.org](#)
- Connect with us on [Slack](#) or via [email list](#)



- [Zowe Github](#)
- [Download Zowe](#)
- Review [documentation](#)
- [Troubleshooting](#)
- Reach out to us on [Slack](#) or via [email list](#)



- Review the [extenders guide](#)
- Zowe [Tutorials](#) and [Samples](#)



- Provide feedback, problems or recommendations to us on [Slack](#) or via [email list](#)
- Submit [Git](#) Issues
- Review the community [backlog](#) and contribute code
- Earn your committer status through [meritocracy](#)

We are building more than just technology, we are building a **community**

Get involved in the Zowe community



Join Open Source Community @

<https://www.openmainframeproject.org/projects/zowe>

Participate in and contribute to the Zowe developer community at zowe.org

Learn how your organization can become a steward and supporter of this project with Open Mainframe Project membership at openmainframeproject.org/about/join





Intro Demo:

<https://www.youtube.com/watch?v=NX20ZMRoTtk&feature=share>

Visual Studio Code (using command line):

https://www.youtube.com/watch?v=la1_Ss27fn8