

# Cloud-Based IDE for Mainframe Development

Dejan Milinkovic  
Broadcom

November 2019  
Session MJ



# Mainframe Challenges

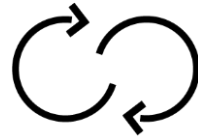
## People



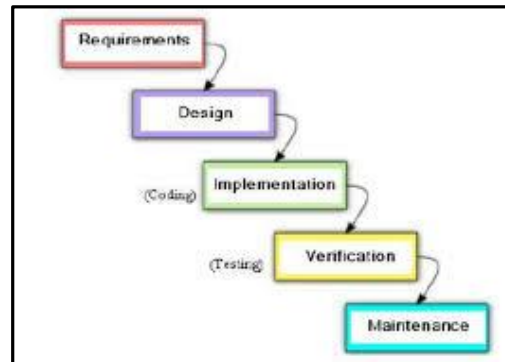
As the mainframe workforce continues to age and retire, a skills shortage is looming



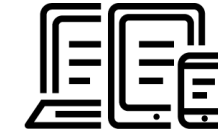
## Process



Old ways of working, once state of the art, are now outdated



## Tools



Traditional tools don't appeal to next gen developers nor provide high productivity

```
Menu Utilities Compilers Options Status Help
ISPF Primary Option Menu
Option ==>
0 Settings      Terminal and user parameters      User ID . : SCOMST0
1 View          Display source data or listings    Time . . : 14:53
2 Edit          Create or change source data        Terminal. : 3277
3 Utilities     Perform utility functions         Screen . . : 1
4 Foreground   Interactive language processing   Language. : ENGLISH
5 Batch        Submit job for language processing   Appl ID . : TSR
6 Command      Enter ISB or Workstation commands   ISB logon : DBPRC9G
7 Dialog Test  Perform dialog testing             ISB prefix: SCOMST0
9 IBM Products IBM program development products   System ID : S001
10 SCLM        SW Configuration Library Manager   MVS acct. : GROUP2
11 Workplace   ISPF Object/Action Workplace       Release . : ISPF 6.1
12 z/OS System z/OS system programmer applications
13 z/OS User    z/OS user applications

Enter X to Terminate using log/list defaults
```

Story to tell

# Mainframe developer tooling

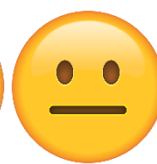
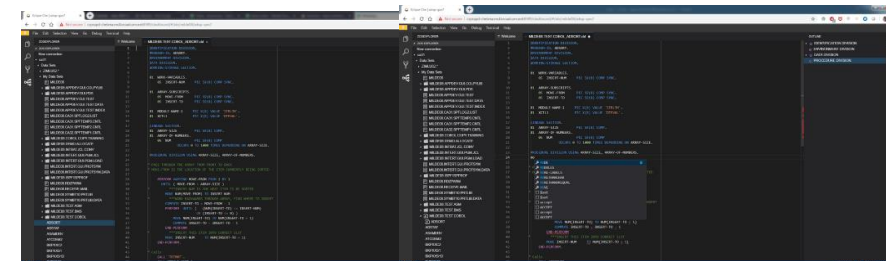
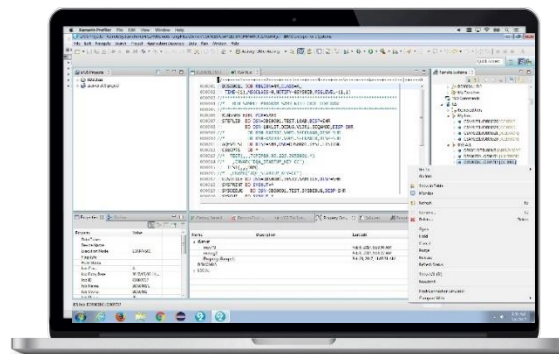
```

Menu Utilities Compilers Options Status Help
-----
ISPF Primary Option Menu

Option ==>

0 Settings Terminal 000100 //IEFBR11 JOB ANS9 ,AVIMASH ,NOTIFY=SYSUDD
1 View 000200 *****
2 Edit Display 000300 //THIS UTILITY USED TO ALLOCATE OR DEALLOCATE A DATASET
3 Edit Catalog 000400 //CREATES BOTH PS AND PDS FILES ALSO DELETES SAME
4 Utilities Perform 000410 *****
5 Foreground Interact 000410 *****
6 Batch Submit j 000420 //MYPROC PROC -----MYPROC IS PROCEDURE NAME THEN PROC THIS IS SYNTAX
7 Command Enter TS 000500 //DELETE EXEC POM=IEFBR14
8 Dialog Test Perform 000600 //SYSOUT DD SYSOUT=*
9 IBM Products IBM prog 000700 //SYSPRINT DD SYSOUT=*
10 SCL 000800 //DD DSN=AN99K ,AVIMASH ,PS4,
11 Workplace 000900 //DISP=(NEW,CATLG,DELETE),
12 z/OS Syst 000910 //DCB=(LRECL=80, BLKSIZE=8000, RECFM=FB),
13 z/OS User 000920 //SPACE=(TRK,(2,3),RLSE)
000930 //
000940 PEND -----END OF PROC.
Enter 000950 *****
000950 //STEP2 EXEC MYPROC -----HERE I AM CALLING MY INSTREAM PROC IN SAME JOB
001000 //SYSIN DD DUMMY
001100 *****
001200 *****
001300 //
001400 *****
001500 // Required TSO segment s/
001600 x = OUTTRAP 'var.1'
001700 address tso "NETSTAT HOME"
001800 parse var var.1 al a2 a3 a4 a5 a6 a7 a8 type .
001900 test = NEWLINE["TCP/IP NAME:" type] ||NEWLINE
002000 IPADDR = SOCKET("GETHOSTID")
002100 parse var IPADDR ip-ic ip-addr
002200 test = text["Connected using IP Address: " ||ip_addr||NEWLINE||NEWLINE
002300 i = 1

```



## Overall Perspective

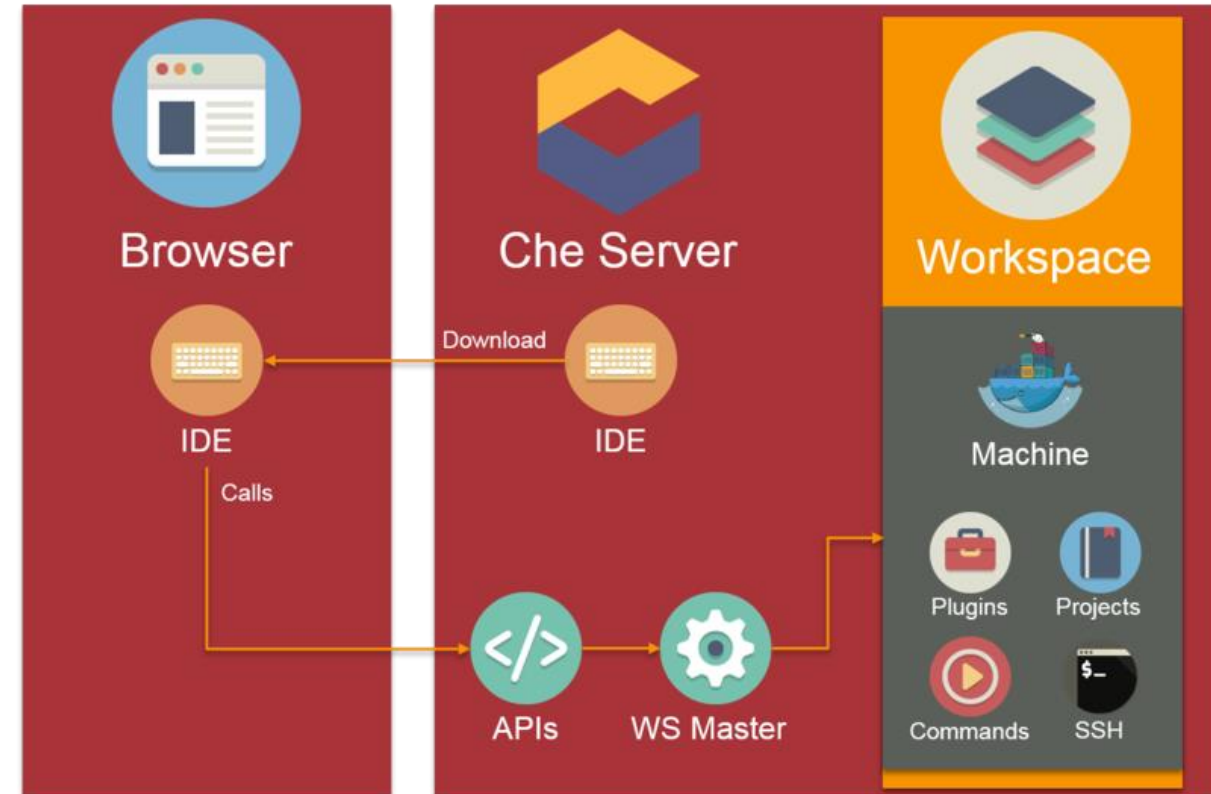
- The next generation of developers wants to use **modern tools**
- The Enterprise prefers to **standardize** on a common stack
- **Leverage heritage**, not to start over
- Embrace **best-of-breed** tooling across platforms, across the Enterprise
- **Open source** to reduce friction when integrating mainframe development across the enterprise
- Mainframe software development should not be **process siloed**

# What is Eclipse Che?



Eclipse Che

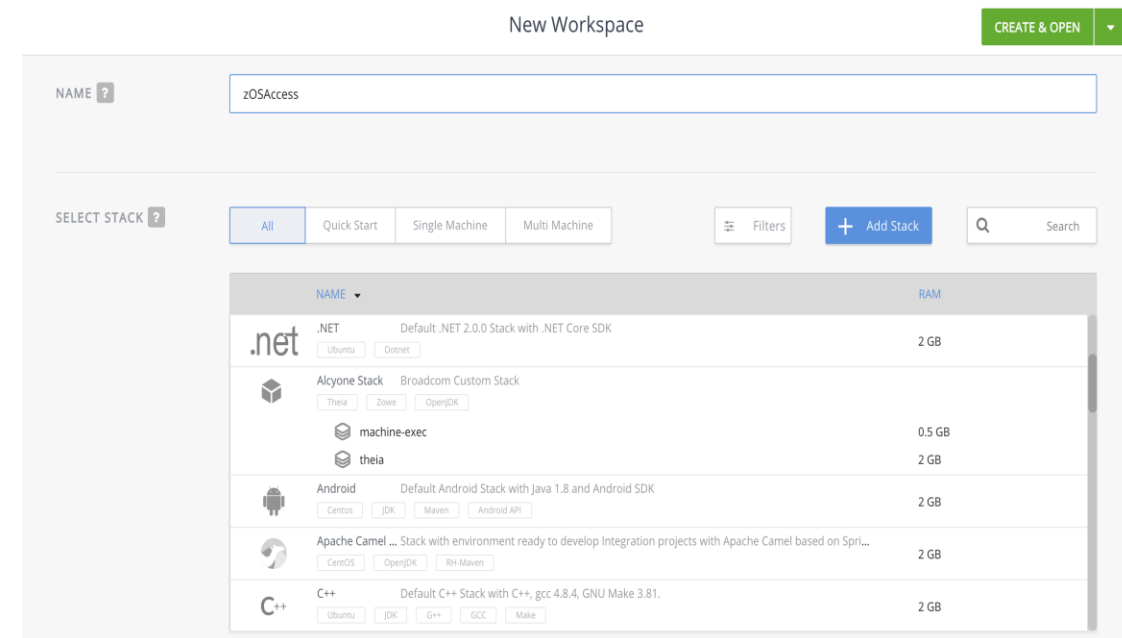
- An **open source Kubernetes-native cloud/hosted IDE** with support for
  - Java, Python, Ruby, PHP, Javascript, C++, JSON, SQL etc.
  - Tools like Git, Subversion, SSH
  - Frameworks like Docker, Openshift, Angular JS etc.
  - Builders like Ant, Maven, Grunt, npm etc.
- Provides **multi-user, team oriented remote development platform**
- Includes a **SDK that allows extensibility to integrate other tools, languages or frameworks**
- Supports **Cloud, On-prem and Hybrid model** of application development
- **Extensible via public APIs**



Eclipse Che workflow

# Why Eclipse Che?

- **Cloud/Web IDE framework**
  - “Bring your own IDE”
  - Support of Language Server Protocol and Debug Adapter Protocol
  - VS Code user experience/extensions support
  - Standard integration with Git
- **Hosted Workspaces**
  - Accelerate project and developer on boarding: zero install development environment
  - Remove inconsistencies between developer environments
  - Built-in security and enterprise readiness
- **Containerized**
  - Easy to install and update
  - Supports running workspaces on Kubernetes, Docker, OpenShift



**But... how is this relevant to us?**



# Our Strategy...

## Open Source-Driven Mainframe development

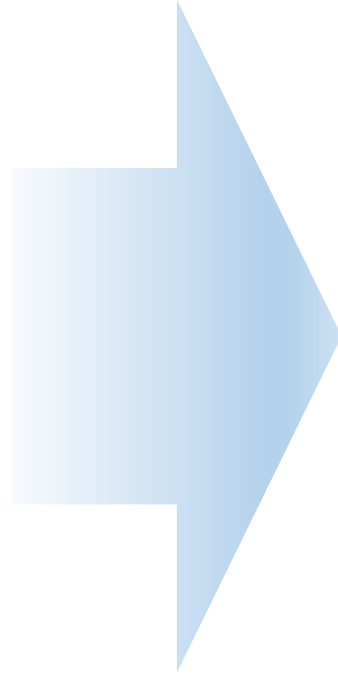


- ✓ What the next-generation knows and wants
- ✓ Enable delivery of modern DevOps built on open source



# Eclipse Che4z Open source subproject

Che extensions for IBM z/OS platform



# Zowe

<https://projects.eclipse.org/proposals/eclipse-che4z>

- Starting with:
  - Connect, authenticate and access mainframe resources – Explorer for z/OS
  - COBOL LSP implementation
- To follow:
  - LSP implementations for other mainframe languages
  - Access to other mainframe resources
  - Community requests

# Eclipse Che4z Open Source Project

## • Status/Roadmap

July 2019

August 2019

September 2019

4Q 2019

- **Announce @ OSCon 2019**

- **Population of the GitHub repos with source code**
  - **Explorer for z/OS**
  - **COBOL LSP**

- **Che4z Beta 1**
  - **COBOL intellisense capabilities**
- **VSCode extensions Beta 1 in marketplace**

- **Che4z Beta 2**
- **VSCode extensions Beta 2 in marketplace**



<https://projects.eclipse.org/projects/ecd.che.che4z>

**What real world use cases will this solve?**



# Different people in company have different goals and challenges

Management



- Cost savings
- Infrastructure maintenance
- 

Administrators



- Deployment
- Centralized Application maintenance
- 

Developers



- No application maintenance
- Modern tooling
-

# Enterprise management

## DevOps Workspace Platform

Manage workspaces at scale with programmable and customizable infrastructure that lets you control system performance, availability, and functionality.

- Use in the cloud or install locally
- Scale horizontally or vertically
- Keep source code off devices
- Enterprise security solutions



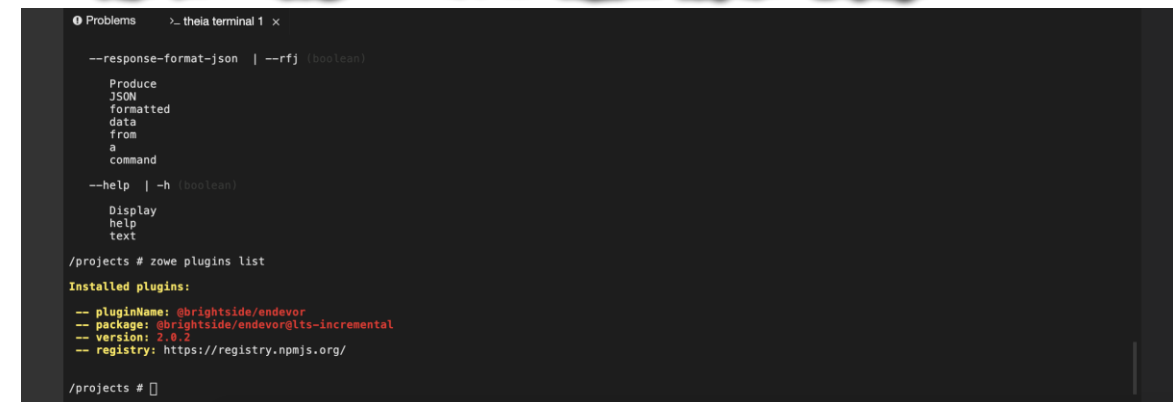
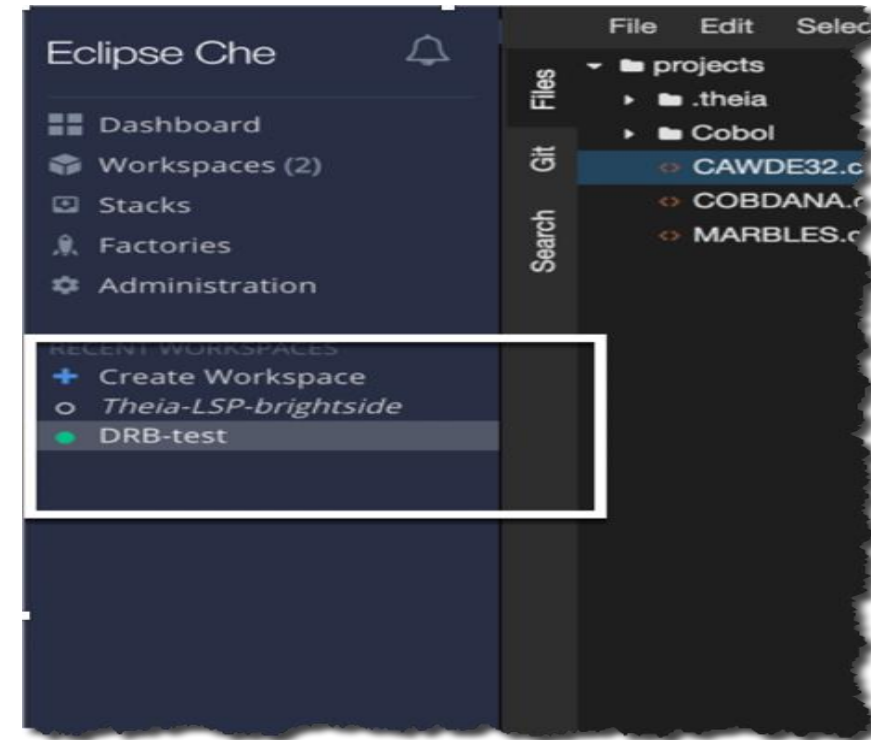
## Quickly get started

### Instant Project Onboarding

- Quickly onboard teams/developers
- Developers in a team can use their local IDE or the Che browser IDE.
  - Share workspaces with anyone
  - Control workspace permissions

### Integrate developer services into a workspace

- Language Servers
- Intellisense and Refactoring
- Debuggers
- **CLI powered by Zowe**

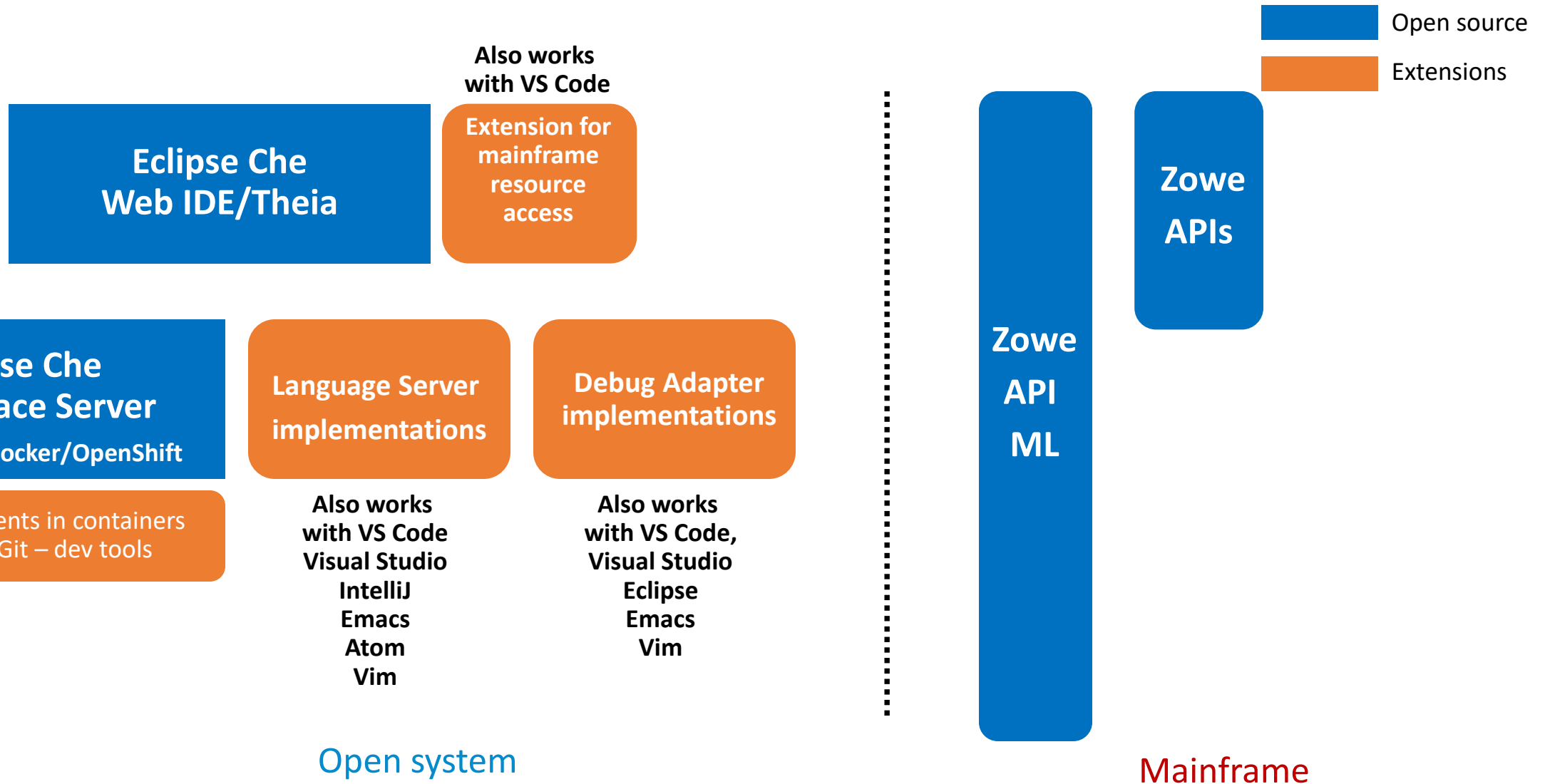


Demo





# The big picture...

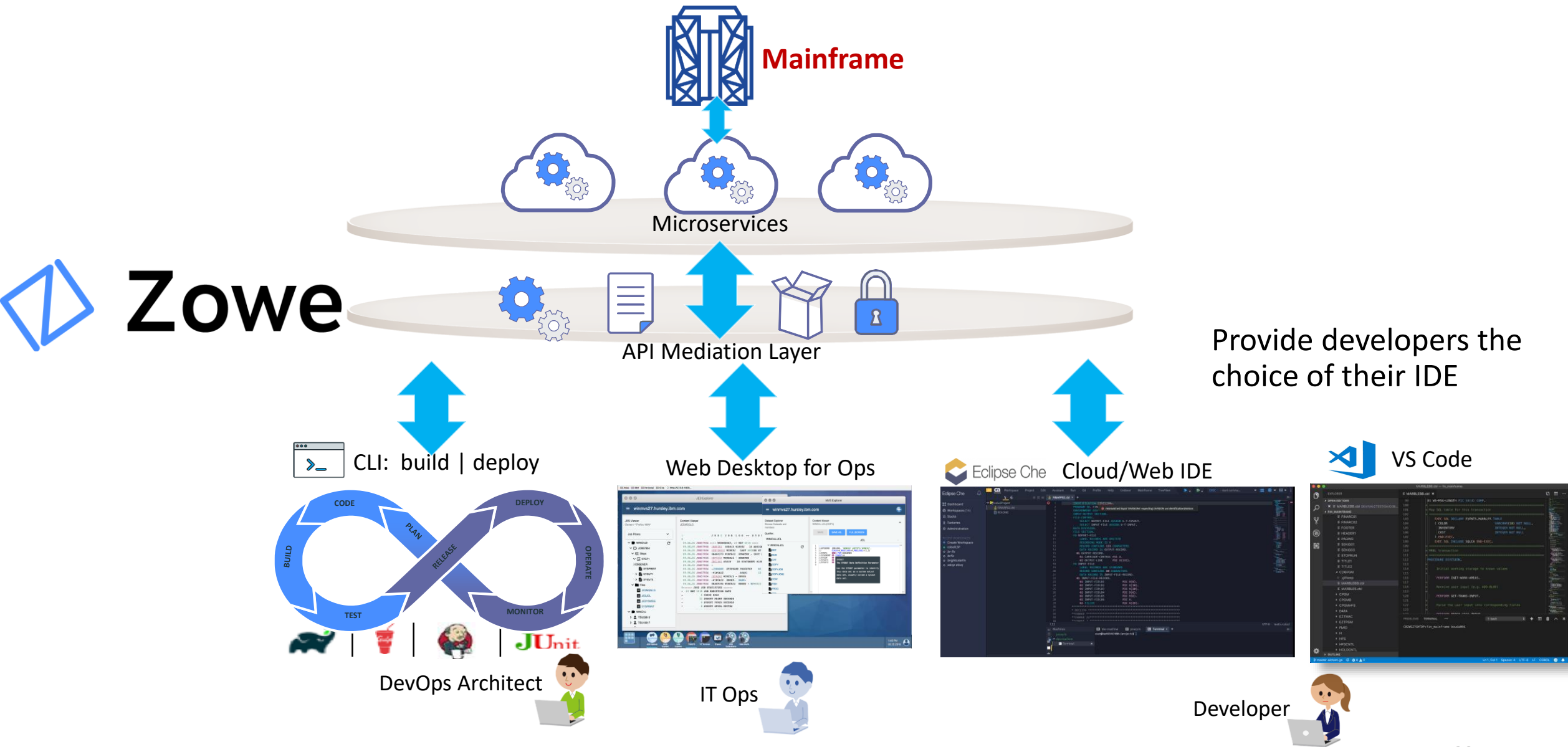


# We are aiming for...

- Flexibility and freedom to developers
  - Allow developers to work with CLI and/or their favorite IDE or Editors, to perform their development tasks with appropriate tooling
- Leverage common components that enable that flexibility and freedom
  - Implementation of standard protocols for editing and debugging
    - Develop once, use in any IDE/Editor that supports the protocol
- Make Mainframe application development like any other platform
  - Centralized management
  - Easy access from anywhere



# Bringing everything together...



Demo






Interested more? -> AO Making the Mainframe a  
Millennial Magnet (wed at 11:45)

Please submit your session feedback!

- Do it online at <http://conferences.gse.org.uk/2019/feedback/nn>
- This session is MJ



1. What is your conference registration number?

 This is the three digit number on the bottom of your delegate badge

--

2. Was the length of this presentation correct?

💡 1 to 4 = "Too Short" 5 = "OK" 6-9 = "Too Long"

1 2 3 4 5 6 7 8 9

3. Did this presentation meet your requirements?

💡 1 to 4 = "No" 5 = "OK" 6-9 = "Yes"

1 2 3 4 5 6 7 8 9

4. Was the session content what you expected?

💡 1 to 4 = "No" 5 = "OK" 6-9 = "Yes"

1 2 3 4 5 6 7 8 9