Intelligent IT Operations

Five Places to Begin Adopting Zowe on Mainframes

ROSE SAKACH

You've heard about Zowe, the new open source software framework designed to make the mainframe an integrated and agile platform within the IT architectural landscape. But where do you actually start? How do you take advantage of the paradigm-changing capabilities Zowe offers?

Practically speaking, you want to proceed in a controlled fashion so you benefit from Zowe without introducing too much change at once. A sound adoption strategy, therefore, is to identify a particular need within your organization and leverage the pieces of Zowe that relate to that need. Focusing your efforts—rather than attempting to "do it all"—will help you build skills around the Zowe platform. As your teams become more skilled with Zowe, you can steadily adopt further capabilities.

Here are five top mainframe needs and the solutions Zowe offers:

1. **Developers Want to Write Code in the Editor of Their Choice**
   Your mainframe developers are likely writing code in ISPF or Eclipse, but those are probably not their tools of choice. Editors such as Visual Studio Code, Visual Studio, Notepad++ and Sublime Text are often more popular options compared to mainframe-native tools like Eclipse.

   Zowe enables developers to use the editor of their choice to write code for the mainframe by increasing source code access. Namely, Zowe enables Git-driven operations (e.g., GitHub, BitBucket or GitLab) to access source code from source code management (SCM) solutions on the mainframe. Git, in turn, gives developers the freedom to use whichever editor they prefer when writing code for the mainframe, dramatically increasing productivity and user satisfaction.

2. **Developers Want the Convenience of Using Task Runners With the Mainframe**
   Task runners simplify and automate common tasks developers need to do as part of writing, building and testing code. Common in the distributed development environment, task runners have traditionally not been used with the mainframe.

   Zowe allows developers to use task runners such as gulp to quickly build automation tools for their own use as well as enable integrations with continuous integration/continuous delivery (CI/CD) tools and integrated development environments (IDEs). For example, through Zowe, developers can use task runners to compile code, link edit code, run batch programs and perform testing—all tasks that would otherwise need to be done manually or through a mainframe-native tool such as Eclipse.

3. **Developers Want Automated Testing Frameworks**
   Unit testing, integration testing, system testing and load/performance testing are all critical when developing code. On the mainframe, most of that testing must be

done manually or through tools such as zUnit or Rexx. It is not uncommon for developers to push testing off to a later development stage, relying on Quality Assurance personnel to conduct testing and catch any problems.

Zowe makes available a whole universe of automated testing frameworks, including Chai, Mocha and JMeter. This enables developers to automate all forms of testing, greatly increasing their efficiency and the quality of their code.

4. **Build and Release Engineers Want Automation of the CI/CD Pipeline**
   Build and release engineers work extensively with traditional mainframe tooling and perform many tasks manually. Through Zowe, however, they can use popular CI/CD tools such as Jenkins, Bamboo and Urban Code in conjunction with task runner tools like gulp to automate common CI/CD tasks. For example, engineers can automate testing, packaging, promoting and deploying code. Such automation of the CI/CD pipeline accelerates the deployment process, minimizes disruptive coding errors, increases overall productivity and lowers costs.

5. **Operations Engineers Want Configuration Management and Infrastructure Orchestration**
   Operations engineers are involved in rolling out software upgrades and application updates. Today, most of this is performed manually. However, modern tools such as Ansible and SaltStack can move these tasks into the streamlined, automated DevOps arena. Zowe enables such tools to be used on the mainframe, delivering version control, configuration consistency and development efficiency— all while boosting speed, minimizing risk and increasing cost savings.

Each of these five needs may exist within your mainframe organization. Even if that is the case, however, pick one to start your exploration of Zowe—then, narrow things down even further by selecting one use case within that need as a pilot project. For example, you may want to automate performance testing of your batch applications or use gulp to automate the task of compiling code. Keep it simple, get the quick win and gain experience. Then, move on to the next need or the next use case.

Modernizing the mainframe to incorporate it into agile DevOps is a process; it will not happen overnight. Shift left, but do so strategically, steadily and systematically. That is the key to success. **ETJ**

Rose Sakach is a Product Management Advisor at Broadcom Inc. responsible for solutions that enable mobile-to-mainframe Software Change Management and DevOps for the enterprise. She has over 30 years of experience in technology roles, including application development, systems programming, operations, product administration, consulting services and product management.
Email: rose.sakach@broadcom.com

Enterprise Tech Journal • 2019: Issue 2 • 17