Introduction to migrating data

Table of contents

1 Migrating data from z/OS to Git

1.1 Introduction to migrating data

Migrating your source code data from z/OS to Git is one of the first steps in implementing a modern CI/CD pipeline. Once you have planned your to-be Git repository layouts, you can use the steps in the following section to migrate your data from z/OS to Git. During the migration process, it is important to be aware of potential issues with migrating non-printable or non-roundtrippable characters, and to have a plan for how to handle them if you encounter them. You can read more about how to handle these characters in Managing code page conversion.

1.2 Steps for migrating data from z/OS to Git

At a high level, the steps for migrating data from z/OS to Git are as follows:

1. Unload the source code from the legacy version control system to PDS libraries: Legacy version control systems often store data in a proprietary format, which IBM's Dependency Based Build (DBB) cannot directly access. Therefore, the source code should first be unloaded to PDS libraries.

2. Load files from PDS libraries to a Git repository on z/OS UNIX System Services (z/OS UNIX) using one of the following methods:

- **DBB migration tool**: The DBB migration tool is provided with DBB, and is the most commonly-used method for migrating data from z/OS to Git.
- **SCLM-to-Git migration tool**: You can use the SCLM-to-Git migration tool if you are moving away from SCLM as your current library manager.
- **Manual migration**: Manual migration of source code from z/OS to Git is possible, but is generally not recommended since it tends to be slower, more tedious, and prone to human error.

3. Once your code is in z/OS UNIX, you can use Git to push it up to your central Git provider.