

Software Configuration Management

Part 1: Configuration-As-Code in DevOps

Define Configuration-As-Code in DevOps

DevOps is a mixture of software development and IT operations designed to reduce the maintenance time for system development. The combination of these practices and tools makes the application of DevOps faster than the traditional software development process [2]. Faster maintenance speeds can increase productivity levels and reduces the overall cost of a project. A definition of configuration as code is scripts or code that include the configurations of the servers or other resources. It is the process of managing configuration files within a repository and controlled with the version control tool [1]. Configuration as code permits changing the settings on an application or product to be tested earlier in the pipeline and completed with greater confidence [1]. Managing the configuration files parallel to the remainder of the code benefits the use of version control for entire projects [5]. The use of configuration as code is beneficial in managing package and component settings [1].

Importance of Configuration-As-Code in DevOps

Configuration-as-code is valuable in the cases of security, traceability, and manageability. Security permits the application of least privilege permissions alongside greater levels of audit access [5]. Any errors traced in a new version are easily reverted to an earlier version because repositories are separate when utilizing version control. Managing configurations as code are beneficial in preventing accidental changes or future damages to the system due to the automated process [5]. Automation of configuration management has time-saving qualities [5]. When setting up or updating configurations, the need for manual dependence and human error is omitted through automation eradicating human error.

Part 2: Support Tools for SCM Functions

Refactoring Tool

Identification entails the selection of a configuration item and records documentation of the functionality and physical characteristics [4]. Ancestry References is an identification support tool used to specify a commit through its ancestry. The GIT tool resolves a reference to the committed parent by inserting a caret at the end of the reference [3]. For example, `$ git show HEAD^` allows viewing of previous commits as the “parent of the HEAD.”

Control entails approval or disapproval of configuration changes of a configured item and establishes repositories to prepare for disaster recovery [4]. Committing your changes is a control tool that allows modified files to be updated and saved (ex. `$ git commit`). Adding `-v` to this example reveals the difference between the changes in the editor to see, showing what changes are being committed [3].

Auditing ensures control of the product evolution from the conceptual to the production phase [4]. Organizations can allow owners access to all information through the Audit Log tab when managing an organization within GitHub. Owners using the Audit Log tab can see what events have occurred at an organization level, who conducted them, and the location of said events [3].

Status accounting entails the record and reporting of information necessary for managing an effective configuration. The significance is for keeping track of information and executing changes [4]. Files within a working directory in a git repository have a state of either tracked or untracked. Tracked files are ones that Git is aware of, while untracked are the remaining files [3]. To check the state of a file, utilize the `git status` command tool (ex. `$ git status`). After a successful run of `git status`, this command tool returns the file status and the current branch [3].

References

1. Configuration as Code: How to Streamline Your Pipeline | Perforce Software. *Perforce Software*, 2020. <https://www.perforce.com/blog/vcs/configuration-as-code#:~:text=Configuration%20as%20code%20is%20the,control%20for%20your%20entire%20product.>
2. What Is DevOps and How Does It Work? | Synopsys. *Synopsys.com*, 2022. <https://www.synopsys.com/glossary/what-is-devops.html>.
3. Git - Revision Selection. *Git-scm.com*, 2022. <https://git-scm.com/book/en/v2/Git-Tools-Revision-Selection>.
4. Collofelo, PhD, James. Software Configuration Management. *Coursera*. 2022
5. Configuration Management In DevOps Practices. Software Testing Help, 2022. <https://www.softwaretestinghelp.com/configuration-management-in-devops/>.