

Journey of Automation - Github, Galaxy, Fedora

Pavel Cahyna, Sergei Petrosian

Automation is more important than ever before in software project management

- Automated Ansible role release and publish to Ansible Galaxy
- Automated Ansible collection build, publish and release to Galaxy
- Automated Fedora RPM build and publish with Packit

Automated GitHub Releases

1. A script that we run manually when we need a new release
2. GitHub action that publishes content of each individual repo into an upstream hub
3. Cron-like daily GitHub Action that collects and publishes content from multiple repositories if any repository has an update

How We Use Conventional Commits

Format: `<type><!>: PR Title`

- **!** - **MAJOR** bump
- `feat` - **MINOR** bump
- `fix`, `ci`, `test`, etc. - **PATCH** bump

feat: Support custom data and logs storage paths

GitHub Action that Processes Conventional PR Titles

1. Figure out the semantic version to use for the new tag
2. Build changelog based on PR types
3. `feat:` -> *New Features*
4. `fix:` -> **Bug Fixes*
5. `else` -> *Other Changes*
6. Pushes a PR with the updated changelog and publishes a new version

Releasing Individual Repositories

GitHub action that publishes repository content to upstream hub once a new version on GitHub is released.

Publishing Content of Multiple Repositories

A cron-like daily GitHub action that and converts multiple repositories into a collection and publishes to upstream Hub if any repository has a new version.

Automated RPM Release with Packit