Automating EPEL branching workflows with ebranch

Michel Alexandre Salim
Production Engineer
michel-slm.name
FAS/IRC/Matrix: salimma

Meta

Agenda

01 What makes EPEL special

02 EPEL workflows

03 Introducing ebranch

04 Examples

05 Future work

06 Resources

07 Q&A

What makes EPEL special?

Fedora, RHEL/CentOS Stream, EPEL

- A subset of Fedora is branched off for CentOS Stream
- RHEL minor releases are cut from CentOS Stream
- Packages in RHEL get official Red Hat support
- Anything else is eligible for EPEL (Extra Packages for Enterprise Linux)
- For the old timers, remember the Fedora Core vs Extras split?

Release Lifecycle

- Fedora releases are maintained for 13 months (2 releases + 1 month)
- CentOS Stream releases are maintained for 5 years
- RHEL gets
  - Full Support: 5 years
  - Maintenance Support: 5 years
  - Extended Life Phase: 2 years

Employee vs volunteer contributor

- Backporting security fixes is not what a Fedora contributor signs up for
- Even for paid employees, it makes sense to limit the scope of what is supported
- Thus Fedora packages not in EL have to be opted into EPEL for each release
- … Dependency hell?

EPEL workflows

CentOS Stream drops

Introducing CentOS Stream 9

EPEL is available

EPEL 9 is now available

Populating EPEL

Get Fedora Package in EPEL

Populating EPEL

- Find out a package you need is not in CentOS Stream
- Request branch / file bug asking maintainers to branch
- Discover it FTBFSes because it needs X missing dependencies
- for i in X; do ...; done

Stale requests

- Many Fedora maintainers are not interested in EPEL
- Most are volunteers so they might not check BZ that often
- For general maintenance, provenpackagers can help
- But branch requests require someone in the ACL
- The EPEL SC created Stalled EPEL Requests for this

Introducing ebranch

A tool for branching Fedora packages for EPEL

pagure.io > epel > ebranch

Thus far my biggest dependency chain requests has gone down 4 levels …

Troy Dawson, EPEL Steering Committee, 2022.01.12 21:30:46

Features

- Calculate recursive dependencies of a package
- Keeps track of previous requests
- TODO File branch requests for them
- TODO Follow up on stale requests

ebranch examples

What BRs are missing?

```
$ ebranch missing-build-reqs python-b4 epel9
{
  "python-b4": {
    "build": {
      "python-dkimpy": [
        "(python3dist(dkimpy) >= 1 with python3dist(dkimpy) < 2)",
        "(python3dist(dkimpy) >= 1.0.5 with python3dist(dkimpy) < 1.1)"
      ],
      "python-patatt": [...] } } }
```

```
ebranch
Usage: ebranch [OPTIONS] COMMAND [ARGS]...

Options:
  --help  Show this message and exit.

Commands:
  build-reqs          lists build requirements for a package
  is-branched         checks if a package is branched
  iterate-report      computes missing BRs for new top-level packages
  ls-branches         lists branches for a package
  missing-build-reqs  lists missing build requirements to build for a branch
  unfold-report       adds new missing BRs to the top-level list
```

Unfold and iterate

```
# add newly discovered BRs to the top-level list
$ ebranch unfold-report b4.json
# mark each you don’t want to follow up on with “skip”: True
# iterate to compute missing BRs for the new top-level packages
$ ebranch iterate-report b4.json epel9
```

Please try and contribute!

```
# enable COPR
$ sudo dnf copr enable salimma/ebranch
# install ebranch
$ sudo dnf install ebranch
# profit!
PSA: EPEL hackfest is right after this session
```

Help needed

- Naming is hard
- Workflow improvement suggestions
- Code contribution
- Packaging improvements
- Will code for beer
- subject to gift limits =)

Future work

Filing branch requests

```
$ ebranch request-branch -f b4.json python-dkimpy epel9
$ ebranch request-missing-build-reqs -f b4.json epel9
```

Dealing with stale requests

```
$ ebranch ping-stale-requests b4.json epel9
$ ebranch escalate-stale-requests b4.json epel9
$ ebranch follow-up b4.json epel9 # do the above two
```

Features

- Parallel build of all the missing BRs
- Detect BRs that are in CentOS Stream but unpublished

Refactor

- Dependency resolution as a standalone module
- [dnf5] use Python API
- Parallel building might make more sense in fedpkg

Resources

- Extra Packages for Enterprise Linux
  - Helping EPEL
  - EPEL Packagers SIG
- ebranch

Questions & Answers


Meta