

# Python packaging in openSUSE

Advantages against other distributions

Matěj Cepl  
mcepl@cepl.eu

openSUSE Conference, June 2022

# Problems in the Python non-specific packaging

- ▶ Multiple interpreters
- ▶ Complicated commands
- ▶ Sheer number of packages
- ▶ Not enough checking

2022-06-02

Python packaging in openSUSE

└─Problems in the Python non-specific packaging

Problems in the Python non-specific packaging

- ▶ Multiple interpreters
- ▶ Complicated commands
- ▶ Sheer number of packages
- ▶ Not enough checking





## Complicated commands

%check

```
%{buildroot}%{python_sitelib}
```

2022-06-02

Python packaging in openSUSE

└─ Complicated commands

And this is still not correct, because it must be done for all Python versions separately (and those `--ignore=build*` must be arranged accordingly). And we need to be able to take into consideration existing `PYTHONPATH`.

```
Complicated commands

%check
export PYTHONPATH=%{buildroot}%{python_sitelib} PYTHONDONTWRITEBYTECODE=1
pytest --ignore=build.* -v
```



## Example

Name

Version

Release

Summary

License

URL

Source

```
# PATCH-FIX-UPSTREAM remove_mock.patch bsc#123456 mcepl@suse.com
```

```
# we don't need stinking mock
```

Patch0

BuildRequires

BuildRequires

BuildRequires

BuildRequires

Requires

BuildArch

%description

```
%define python_module() python3-%{**}}
```

```
%{version}
```

2022-06-02

## Python packaging in openSUSE

Example

1. Eliminate as much boiler-plate as possible. We have only two lines now, which are same every time: that `python_module` definition and `Release`. And in both cases we are just forced to have them there by the mechanics of RPM.
2. Note that macro `python_subpackages`, that's the root of all machinations. In effect this SPEC file is just a foundation, I don't want to use the term "template", because that would be misleading, for multiple generated ones.

Example

```
{%python_module:do} python_module() python3-%{**}
Name: python-fisher
Version: 1.0.2
Release: 0
Summary: Example Fisher package
License: MIT
URL: https://github.com/mamba/fisher
Source: https://files.pythonhosted.org/packages/source/F/fisher/fisher-%{version}.tar.gz
# do not need stinking mock
Patch0: remove_mock.patch
BuildRequires: %python_module applydiff
BuildRequires: %python_module pytest
BuildRequires: setuptools
Requires: python-egginfo
BuildArch: noarch
%python_subpackages

Description:
Fisher is a collection of enhancements to the Python packages that
allow you to build and distribute Python packages.
```

## Example (cont.)

`%prep`

`%{version}`

`%build`

`%install`

`%{buildroot}`

`%check`

`# %pyunittest discover -v`

`%files`

`%doc`

`%{python_sitelib}`

`%{python_sitelib}    %{version}`

`%changelog`

2022-06-02

## Python packaging in openSUSE

└─ Example (cont.)

1. Using autosetup to avoid dealing with each patch again.

Example (cont.)

```
Specfile
Autosetup -p1 -s /usr/share/autosetup
Source
export SOURCE_DIR=python_build
Install
python_install
python_sitelib %{buildroot}/%{python_sitelib}
Check
# %pyunittest discover -v
python_test
Files
python_files
License LICENSE
Doc CHANGES.txt README.txt
python_sitelib %{buildroot}/%{python_sitelib}
python_sitelib %{buildroot}/%{python_sitelib}-%{version}-info
Changelog
```





Thank you!

2022-06-02

## Python packaging in openSUSE

└─ Thank you!

Thank you!

My email is [mcp1@cepl.eu](mailto:mcp1@cepl.eu).

There is also interesting presentation by my colleague Steve Kowalik on the Python macros at <https://youtube.com/95Dfkw1s2IA>

The source of this presentation is available on <https://github.com/mcp1/opensusemlgpt>.