Hunter Bot Documentation

Release 1.0

Alex Frappier Lachapelle

Sep 27, 2017

Contents

This documentation is about starting and using bot which will help to do testing and automatically releases creation during development process of Hunter package manager.

CHAPTER 1

Installing Rust

See also:

- Official way to install Rust
- Uninstalling Rust
- Downloads

To install Rust from `.tar.gz` archives you need to download `rust` and `rust-std` archives and merge them:

```
> wget https://static.rust-lang.org/dist/rust-std-1.10.0-x86_64-unknown-linux-gnu.tar.
↪gz
> tar xf rust-std-1.10.0-x86_64-unknown-linux-gnu.tar.gz

> wget https://static.rust-lang.org/dist/rust-1.10.0-x86_64-unknown-linux-gnu.tar.gz
> tar xf rust-1.10.0-x86_64-unknown-linux-gnu.tar.gz

> mv rust-std-1.10.0-x86_64-unknown-linux-gnu/rust-std-x86_64-unknown-linux-gnu/lib/
↪rustlib/x86_64-unknown-linux-gnu rust-1.10.0-x86_64-unknown-linux-gnu/rustc/lib/
↪rustlib
```

Now add `rustc` and `cargo` to `PATH`:

```
> export PATH=/.../rust-1.10.0-x86_64-unknown-linux-gnu/rustc/bin:$PATH
> export PATH=/.../rust-1.10.0-x86_64-unknown-linux-gnu/cargo/bin:$PATH
```

Check tools can be found:

```
> which cargo
/.../rust-1.10.0-x86_64-unknown-linux-gnu/cargo/bin/cargo
> cargo --version
cargo 0.11.0-nightly (259324c 2016-05-20)

> which rustc
/.../rust-1.10.0-x86_64-unknown-linux-gnu/rustc/bin/rustc
> rustc --version
rustc 1.10.0 (cfcb716cf 2016-07-03)
```

Run build:

```
> cargo build --verbose
```

See also:

- Ansible role

CHAPTER 2

Install

CHAPTER 3

Build documentation

Documentation build by sphinx and hosted on readthedocs service.

See also:

- Good tutorials

Local

To build documentation on local machine use virtualenv:

```
> cd docs
[docs]> virtualenv _venv
[docs]> source _venv/bin/activate
(_venv) [docs]> which pip
/.../docs/_venv/bin/pip
```

Install dependencies:

```
(_venv) [docs]> pip install -U pip
(_venv) [docs]> pip install -r requirements.txt
```

Build documentation:

```
(_venv) [docs]> mkdir _static # if not exists
(_venv) [docs]> sphinx-build -v -W . _build
```

Open `index.html` in browser:

```
(_venv) [docs]> ls _build/index.html
_build/index.html
```

Run spell checker:

```
(_venv) [docs]> sphinx-build -b spelling . _spelling
```

Helper scripts

Alternatively you can use `jenkins.sh` and `make.sh` scripts.

To initialize environment, build documentation and run spell checker at once:

```
> cd docs
[docs]> ./jenkins.sh
```

Do the same and stay in environment:

```
> cd docs
[docs]> source ./jenkins.sh
(_venv) [docs]> which pip
/.../docs/_venv/bin/pip
```

`jenkins.sh` will build documentation from scratch. This may be useful in case some HTML/CSS files stuck in temporary directory. To update documentation without deleting old files use `make.sh` script (which is usually much faster):

```
(_venv) [docs]> ./make.sh
...
Done:
/.../docs/_build/index.html
```