

Nov 22, 19 8:43

**Makefile**

Page 1/2

```

1 VERSION      = 1.6.3
2 RELEASE      = 2
3 DATE         = $(shell date)
4 NEWRELEASE   = $(shell echo $$(( $(RELEASE) + 1 )) )
5 PYTHON       = python
6 PROJECT_NAME = profile_projects
7 TOPDIR       = $(shell pwd)
8 DIRS         = profiles_api profiles_project
9 PYDIRS       = profiles_api profiles_project
10 EXAMPLEDIR   = examples
11 MANPAGES     =
12 A2PS2S1C     = /bin/a2ps --sides=2 --medium=Letter --columns=1 --portrait --line-numbers=1 --font-size=8
13 A2PSTMP      = ./tmp
14 DOCS         = ./docs
15
16 SHELL := /bin/bash
17
18 #all: rpms
19 all: pdf
20
21 #https://stackoverflow.com/questions/6273608/how-to-pass-argument-to-makefile-from-command-line
22 args = `arg="$(filter-out $@,$(MAKECMDGOALS))" && echo ${arg:-${1}}`
23 %:
24     @:
25
26 versionfile:
27     echo "version:" $(VERSION) > etc/version
28     echo "release:" $(RELEASE) >> etc/version
29     echo "source build date:" $(DATE) >> etc/version
30
31 manpage:
32     for manpage in $(MANPAGES); do (pod2man --center=$$manpage --release="" ./docs/$$manpage.pod > ./docs/$$manpage.1); done
33
34
35 build: clean
36     $(PYTHON) setup.py build -f
37
38 clean: cleantmp
39     -rm -f MANIFEST
40     -rm -rf dist/ build/
41     -rm -rf *~
42     -rm -rf rpm-build/
43     -rm -rf deb-build/
44     -rm -rf docs/*.1
45     -rm -f etc/version
46     -find . -type f -name *.pyc -exec rm -f {} \;
47     -find . -type f -name *~ -exec rm -f {} \;
48     -find profiles_* -type f -name "__init__.py" -exec rm -rf {} \;
49     -find profiles_* -type d -name "__pycache__" -exec rm -rf {} \;
50
51 clean_hard:
52     -rm -rf $(shell $(PYTHON) -c "from distutils.sysconfig import get_python_lib; print get_python_lib()") /adagios
53
54
55 clean_hardest: clean_rpms
56
57
58 install: build manpage
59     $(PYTHON) setup.py install -f
60
61 install_hard: clean_hard install
62
63 install_harder: clean_harder install
64
65 install_hardest: clean_harder clean_rpms rpms install_rpm
66
67 install_rpm:
68     -rpm -Uvh rpm-build/adagios-$(VERSION)-$(NEWRELEASE)$(shell rpm -E "%{?dist}") .noarch.rpm
69
70
71 rebuild: install_harder
72
73 clean_rpms:
74     -rpm -e adagios
75
76 sdist:
77     $(PYTHON) setup.py sdist
78
79 pychecker:
80     -for d in $(PYDIRS); do ($(MAKE) -C $$d pychecker ); done
81 pyflakes:
82     -for d in $(PYDIRS); do ($(MAKE) -C $$d pyflakes ); done
83
84 money: clean
85     -sloccount --addlang "makefile" $(TOPDIR) $(PYDIRS) $(EXAMPLEDIR)

```

Nov 22, 19 8:43

## Makefile

Page 2/2

```

86
87 testit: clean
88     -cd test; sh test-it.sh
89
90 unittest:
91     -nosetests -v -w test/unittest
92
93 rpms: build sdist
94     mkdir -p rpm-build
95     cp dist/*.gz rpm-build/
96     rpmbuild --define "_topdir %(pwd)/rpm-build" \
97     --define "_builddir %{_topdir}" \
98     --define "_rpmdir %{_topdir}" \
99     --define "_srcrpmdir %{_topdir}" \
100     --define '_rpmfilename %{NAME}-%{VERSION}-%{RELEASE}.%{ARCH}.rpm' \
101     --define "_specdir %{_topdir}" \
102     --define "_sourcedir %{_topdir}" \
103     -ba adagios.spec
104 debs: build sdist
105     mkdir -p deb-build
106     cp dist/*gz deb-build/adagios_${VERSION}.orig.tar.gz
107     cp -r debian.upstream deb-build/debian
108     cd deb-build/ ; \
109     tar -zxvf adagios_${VERSION}.orig.tar.gz ; \
110     cd adagios-${VERSION} ;\
111     cp -r ../debian debian ;\
112     debuild -i -us -uc -b
113
114 coffee:
115     cd adagios/media/js/ && coffee -c adagios.coffee
116
117 trad: coffee
118     cd adagios && \
119     django-admin.py makemessages --all -e py,html && \
120     django-admin.py makemessages --all -d djangojs && \
121     django-admin.py compilemessages
122
123 #Ref: https://stackoverflow.com/questions/1490949/how-to-write-loop-in-a-makefile
124 # MANIFEST
125 SRC1= Makefile setup.py README.rst example.py setup.cfg
126 SRC2= tox.ini
127 #SRC2= manage.py profiles_projects-dir-layout.txt
128
129 cleantmp:
130     rm -f ${A2PSTMP}/*.ps ${A2PSTMP}/*.pdf
131 ps: cleantmp
132     $(foreach var, $(SRC1), ${A2PS2S1C} $(var) --output=${A2PSTMP}/${var}.ps ;)
133     $(foreach var, $(SRC2), ${A2PS2S1C} $(var) --output=${A2PSTMP}/${var}.ps ;)
134
135 allpdf: pdf
136     make -C profiles_api pdf
137     make -C profiles_project pdf
138
139 pdf: ps
140     $(foreach var, $(SRC1), (cd ${A2PSTMP};ps2pdf $(var).ps $(var).pdf);)
141     $(foreach var, $(SRC2), (cd ${A2PSTMP};ps2pdf $(var).ps $(var).pdf);)
142     rm -f ${A2PSTMP}/*.ps
143     cp ${A2PSTMP}/*.pdf ${DOCS}/
144
145 tree: clean
146     tree -L 4 > ${PROJECT_NAME}-dir-layout.txt
147
148 # https://stackoverflow.com/questions/7507810/how-to-source-a-script-in-a-makefile/7508273
149 activate:
150     @echo "Run command"
151     @echo "source ./env/bin/activate"
152     @echo " deactivate to exit"
153
154 status:
155     git status
156 commit:
157     git commit -am "$(call args,Automated lazy commit message without details, read the code change)" && git push
158
159 venv:
160     source venv/bin/activate
161     pip install -e . # To create an editable install of this package
162     pip install --upgrade pip
163     # https://github.com/ManageIQ/manageiq-api-client-python
164     # $ export MIQURL=http://localhost:3000/api
165     # $ export MIQTOKEN=< place real token here >

```