

# Meta Test Family

-

# Testování modulů a kontejnerů

Senior Software Engineer, Red Hat s.r.o.

Petr Hráček, OpenAlt 2017

4/11/2017

# Agenda

- Proč používat MTF?
- Základní informace
- Vlastnosti
  - Test docker container
  - Test module
- Automation
  - CI pro MTF
  - Reálné použití MTF a plány
- Backends (conu, nspanw)

# PROČ?



# INFO

# Kde můžeme použít MTF?

- Testování modulů
- Testování kontejnerů
- Tzv. lintery definované MTF - **linters DEMO**



## Kde není MTF použitelné?

- MTF není CI. To je Vaše volba, vyberte si co chcete
  - Jenkins
  - Taskotron
- MTF není testovací framework, vyberte si sami
  - Poskytuje přímou podporu pro *avocado*
  - Příklady pro *behave*, *unittest*, *pytest*

# Základní informace

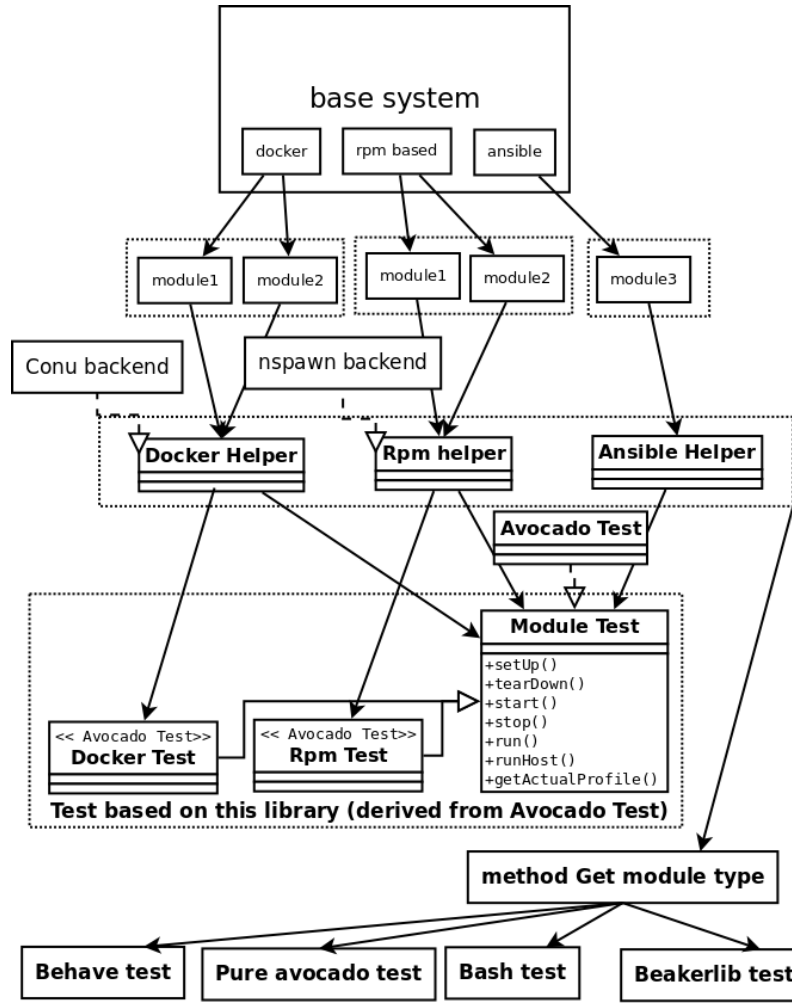
- **Meta-Test-Family (MTF)**
- MTF připraví prostředí
  - závislosti pro MTF
    - Docker, Rpm, nspawn
    - Nastartuje služby
  - testovací závislosti, které jsou nutné pro hosta
- MTF spustí testy bez přípravy prostředí

# Konfigurace

- Pomocí config.yaml souboru
  - memcached ([examples/memcached/config.yaml](#))
  - celý s komentáři ([docs/example-config.yaml](#))
  - Spuštění bez config.yaml ([docs/examples-config-minimal.yaml](#))
  - Dokumentace (RTD: [user\\_guide/how\\_to\\_write\\_conf\\_file.html](#))
- Pomocí proměnný prostředí
  - Lze předefinovat v config.yaml
  - Dokumentace (RTD: [user\\_guide/environment\\_variables.html](#))



# Základní struktura



Whole system - host  
various modules

Module types

Low level handling libraries

MTF - helper layer - read configs  
improve backend handling

Avocado Test object

Avocado Test Inherited classes  
for various artifacts

Low level handling of MTF  
if you would like to use other frameworks  
instead of avocado

# Jak nainstalovat MTF?

- Instalace z našeho COPR repositáře  
<https://goo.gl/WBtsYa>
  - `dnf copr -y enable phracek/meta-test-family-devel`
  - `dnf install -y meta-test-family`
- Stabilní verzi z Fedory
- Stabilní repositář z COPR repositáře
  - `dnf copr -y enable phracek/meta-test-family`
  - `dnf install -y meta-test-family`
- From [source code](#) via `sudo make install`

# FEATURES

## MTF a příprava prostředí

- **Příprava prostředí pro testy**

```
sudo MODULE=docker mtf-env-set
```

- **Spust'me testy**

```
sudo MODULE=docker mtf *.py
```

- **Obnova prostředí před testy**

```
sudo MODULE=docker mtf-env-clean
```

## MTF bez přípravy prostředí

- `sudo MODULE=docker mtf *.py`

## Testování dovecotu

- `sudo MODULE=docker mtf-env-set`
- `sudo MODULE=docker mtf -l *.py`
- `sudo MODULE=docker mtf-env-clean`

# Testy v nspawn prostředí

- Nastavení proměnných modulemd-url v config.yaml  
moduly
- Používá systemd-nspawn
- Příkazy:  
sudo MODULE=nspawn mtf-env-set  
sudo MODULE=nspawn mtf -l \*.py
- **DEMO**

# AUTOMATION



# MTF CI

- Testing module - check\* v Makefile
  - [examples/testing-module/Makefile](#)
- Travis
  - Pouze docker testcases, (ubuntu trusty), rychlé

# Reálné příklady použití MTF

- Testování modulů
  - Taskotron přes [task](#)
- Testování kontejnerů (workaround pro buildování kontejnerů)
  - Taskotron (čeká se na RFE z [taskotronu](#))

# DEMO

LINK

# BACKENDY

## Backendy, Budoucnost

- Zájem o low-level knihovnu místo MTF
- **Conu** - CONTainer utils, oddělený projekt [link](#)
- Will be easier to create new backends (**openshift, atomic**)

# Linky

- <http://meta-test-family.readthedocs.io/en/latest/index.html>
- <https://github.com/fedora-modularity/meta-test-family>
- <https://copr.fedorainfracloud.org/coprs/phracek/meta-test-family/>
- <https://copr.fedorainfracloud.org/coprs/phracek/meta-test-family-devel/>
- <https://github.com/fedora-modularity/conu/>
- **MTF team**
  - Irina Gulina, Jan Ščotka, Petr Hráček, Petr Sklenář

# QUESTIONS?