Avocado: Open Source Testing Made Easy
LinuxCon North America, 2015

August 20th, 2015

Cleber Rosa <cleber@redhat.com>
Software Engineer
Agenda

● What is Avocado?
● Architecture
● Features
● Demo
● Roadmap
Who we are

- Virtualization Test Team @ Red Hat
- We develop testing tools for KVM and Libvirt
- We maintain Autotest and virt-test
- The experience with those prompted us to imagine what the next generation of testing tools would look like
Without further ado...

- **Avocado** is a set of tools and libraries to **perform automated testing on linux platforms**
- Developed to reconcile the needs of different teams involved in software development: **QE** and **Development**
Testing tools: QE vs. Development

Let's crash and burn it! if we can't crash it, we're not working hard enough

In code we trust (Works For Me) we write good code – and keep getting better at it
Avocado: A new testing toolbox

Testing should be fun and simple:

- Start with a test runner, with optional features helpful for debugging and development
- Add more building blocks (plugins) as you need more features
- Don't restrict test development choices - Use any language you want (you get benefits from using test APIs though)
- The same test runner is used in the infrastructure that runs CI jobs - the test grid
Avocado For Test Writers

Test Runner

simple tests

Testing API

Instrumented tests

GDB Wrappers

Program under test
Multiplexer

- In virt testing, we have large test matrixes
  - Disk formats
  - NICs
  - Guest OS
  - Host OS
- The multiplexer is a mechanism of describing a test matrix in a compact way
- YAML based
- Allows the use of filters to reduce the scope of the matrix
Multiplexer - simple example

```plaintext
bread: !mux
  italian:
    bread: Italian
  nine_grain:
    bread: Nine grain
topping: !mux
  american:
    topping: American
  cheddar:
    topping: Cheddar
filling: !mux
  roast_beef:
    filling: Roast Beef
tuna:
  filling: Tuna
```

Variants generated:

1. /bread/italian,/topping/american,/filling/roast_beef
2. /bread/italian,/topping/american,/filling/tuna
3. /bread/italian,/topping/cheddar,/filling/roast_beef
4. /bread/italian,/topping/cheddar,/filling/tuna
5. /bread/nine_grain,/topping/american,/filling/roast_beef
6. /bread/nine_grain,/topping/american,/filling/tuna
7. /bread/nine_grain,/topping/cheddar,/filling/roast_beef
8. /bread/nine_grain,/topping/cheddar,/filling/tuna
Multiplexer - complex example

73 line YAML file

Tree representation

1440 variants
Real World: Jenkins Integration
Real World: Jenkins Integration
Avocado: Future

- More external contributions
- Improve virtualization support
- Integrate with more CI tools and provisioning tools
- Avocado server reports and REST tools
- Component isolation (automated bisection)
- ... You decide!
Resources

● Main website
  ○ http://avocado-framework.github.io/

● Documentation

● COPR repo
  ○ https://copr.fedoraproject.org/coprs/lmr/Autotest/