VirtualBox

VirtualBox is an open source and free virtualization software platform for Windows, Mac, Linux, and other operating systems. It's a great fit for Vagrant and development environments because it's free and consistent in usage. For networking, it's also a great fit, as you can name the virtual network segments. VMware is faster for desktop virtualization, but costs money, has two separate versions (Workstation or Fusion) that operate a little differently, and requires a paid plugin to work with Vagrant.

Vagrant

Vagrant is a tool that takes the pain out of creating and using development environments. In the past, users used their own PC, procured a server, or used a virtual machine on their desktop or a server. Over the last decade, the virtualization revolution has dictated that these are most-commonly virtual machines (VMs). Using VMs leverages the modern hardware on your PC that could duplicate the power of a rack of computers from years ago, but let's look at the few of things that Vagrant helps with:

- **Consistent**: VMs are snapshot installs of an installed OS, but Vagrant can take this to the next level across complex environments of multiple VMs. Your environment and mine will always be the same.
- **Resilient**: your development environment can now be infinitely reinstalled. Of course, you can reinstall a VM, but with Vagrant, you type `vagrant destroy`, and `vagrant up`. If your data resides somewhere else, Vagrant can use shared folders on your host machine to keep working data unaffected.
- **Portable**: your Vagrant Box (a VM that's been "Vagrantized") can be shared with anyone via a URL or Atlas. You can also give someone a Vagrantfile, which is a simple text file, and they can type `vagrant up` to launch your environment!
- **Shareable**: You can share your running instance with another user with Vagrant Share. This is great for classes and troubleshooting.

PyEZ Library

PyEZ (TechWiki)(GitHub)(PyPI) is Juniper's premiere NETCONF-based automation library. Originally authored by Junos Automation Hall of Famer Jeremy Schulman, PyEZ (aka junos-eznc for "EZ" NETCONF) is now curated by another automation rockstar, Rick Sherman. With thousands of downloads a month, the Python library is definitely the hottest open source network automation library out there!

Before PyEZ (and our other EZ libraries), network automation usually meant programming, but PyEZ helps make this consumable for non-programmers.

Python as a Power Shell techwiki page:

> This means that non-programmers, for example, the Network Engineer, can use the native Python shell on their management server (laptop, tablet, phone, and so on) as their point-of-control for remotely managing Junos OS devices. The Python shell is an interactive environment that provides the necessary means to perform common automation tasks, such as conditional testing, for-loops, macros, and templates. These building blocks are similar enough to other "shell" environments, like Bash, to enable the non-programmer to use the Python shell as a power tool, instead of a programming language. From the Python shell, a user can manage Junos OS devices using native hash tables, arrays, and so on, instead of using device-specific Junos OS XML or resorting to "screen scraping" the actual Junos OS CLI.

Most network vendors don't have truly open libraries that are free as in freedom, vs. free as in beer. You might have to sign up on a webpage, fill out a form, or actually buy something. Juniper's commitment to real open source projects and tooling is best presented with our GitHub pages (Juniper) (JNPRAutomate).

Ansible

You hear a lot about Puppet, Chef, Ansible, and Salt, but Ansible definitely wins the award for easiest configuration management framework to learn. That doesn't mean it's less powerful, as it's currently powering some of the largest clouds in the world. Ansible's strengths for us:

- rapid configuration of multiple devices (up to thousands), simultaneously over SSH
- easy configuration through use of YAML for describing your environment
- easy templating with Jinja2 to dynamically
- easy operation with CLI command suite Swiss army knife tools, `ansible` and `ansible-playbook`
- easy module installation with `ansible-galaxy`

Did we mention it's easy? Ansible will make your day-to-day job easier, with immediately beneficial network automation prowess. After checking out Ansible, you'll be that much more prepared to learn Puppet, Chef, and Salt and see if they're a better fit for you.