Why use Ansible?

Ansible is agentless

Ansible doesn't need any agents to be installed on remote systems to be managed, which means less maintenance overhead and performance issues. Instead, Ansible uses a push-based approach leveraging existing SSH connections to run tasks on the remote managed host. Chef or Puppet work by installing an agent on the hosts to be managed and the agent pulls changes from the control host using its own channel.

It's written in Python

Ansible is written in Python, which means installing and running Ansible in any Linux distribution is very easy, and only a little more difficult on OS X. Being a popular language, there's also a good chance that you're familiar with it, or at least can find enough resources online to start learning. Or, you'll easily be able to find a developer with Python experience to help you out.

Learn Ansible in minutes

The fact that a new user can get up to speed and run Ansible tasks in a matter of minutes, thanks to clear and easy-to-follow documentation, is one of the most appealing features of Ansible. Troubleshooting in Ansible is also very easy for beginners, and the fact that all tasks are idempotent reduces the risk of making a mistake.

Deploy infrastructure in record time

Ansible can dispatch tasks to multiple remote managed hosts in parallel. This means you can execute Ansible tasks on a second managed host without waiting for them to complete on the first to reduce provision time and deploy your infrastructure faster than ever.