Ansible

A tool that helps you automate IT tasks.

* Execute tasks from your own machine remotely.
* configuration / installation / deployment steps in a single YAML file
* Re-use same file multiple times and for different environments.
* More reliable and less error prone.

**Ansible is agentless**Ansible does not need anything installed in the target machines.

Ansible works with modules. Modules are small programs that do the actual work. They get send to the control machine to the target server... do their job and then are removed.  
  
Examples of modules are:

* Start Docker Container
* Create or copy a file
* Install Nginx Server

The list of modules can be seen in the Ansible official documentation.  
  
Ansible uses simple YAML language, it is upper intuitive.

**Ansible Playbooks**  
  
Multiple modules in a certain sequence grouped together.  
  
Sequential modules are grouped in tasks. Where each task makes sure the module gets executed with certan arguments and also describes a task using a name.  
  
Example:

- hosts: databases 🡺 where should these tasks be executed  
 remote\_user: root 🡺 which user will be used  
 vars:  
 tablename: foo

tableowner: someuser

tasks:  
 - name: Rename table foo to bar  
 postgresql\_table:  
 table: {{ tablename }}

rename: bar  
  
- name: Set owner to someuser  
 postresql\_table:  
 name: {{ tablename }}

owner: {{ tableowner }}

**Play:**  
  
Define which tasks, in which hosts, by whch users.  
  
**Playbook:**  
  
Orchestrates the module execution. A playbook is a file.  
In a file there can be many Plays defined.  
  
The attribute “name” is used to uniquely identify its purpose.   
  
  
**Hosts (-hosts)**  
  
In Ansible there is a hosts file that keeps a list of inventories (all the machines involved in Ansible tasks execution).  
So this attribute is mapped to the values inside this hosts file.  
  
The hosts file looks like:

10.24.0.100  
  
[webservers]  
10.24.0.1  
10.24.0.2  
  
[databases]  
10.24.0.7  
10.24.0.8

With Ansible we can create an alternative of Docker file … and do it in a more generic way that allows us to reproduce those steps on many environments.  
  
You can manage not only the Docker Container but the Host / Storage / Network where the Docker container is running.  
  
**UI dashboard from Red Hat**  
  
Centrally store all the automation tasks across teams .. grant permissions for those teams … manage inventory (which things have run and their status).  
  
**Advantages of Ansible over Pupped and Chef**  
1. Ansible users simple YAML, whereas the others user Ruby (you need to learn a new language).  
  
2. Ansible is Agentless (no installation on target servers is needed), whereas the others require installation on target machines (and this means a maintenance cost for updating / upgrading etc … ).