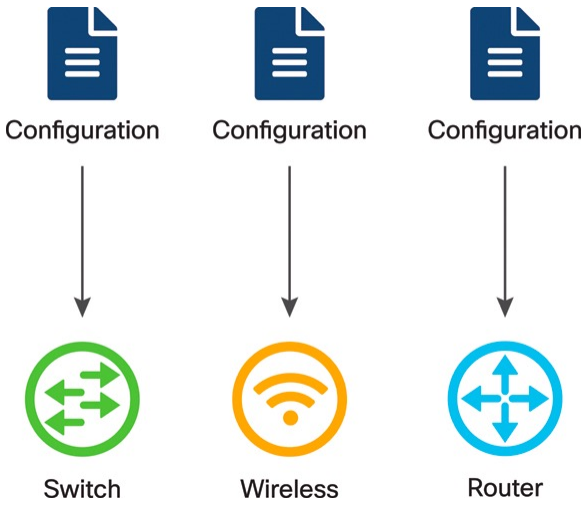
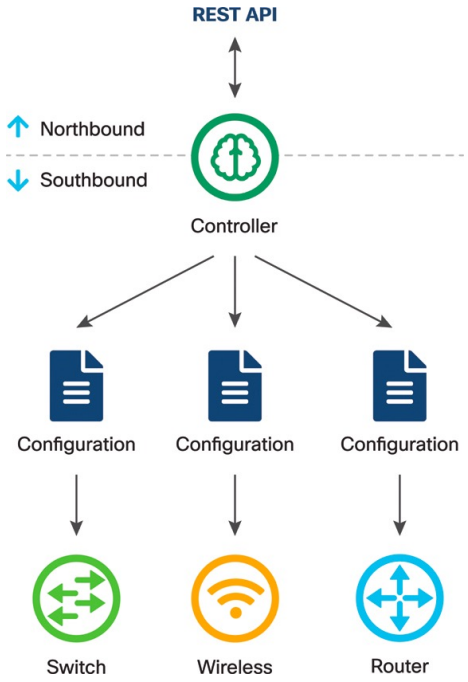
CONTROLLER VERSUS DEVICE-LEVEL MANAGEMENT

CONTROLLER VERSUS DEVICE-LEVEL MANAGEMENT

- *Network Managed on a Device-by-Device Basis*



- *Network Managed Through a Network Controller*



AUTOMATION TOOLS

Common Automation Frameworks

Agent Based

- Puppet
- Chef

Agentless

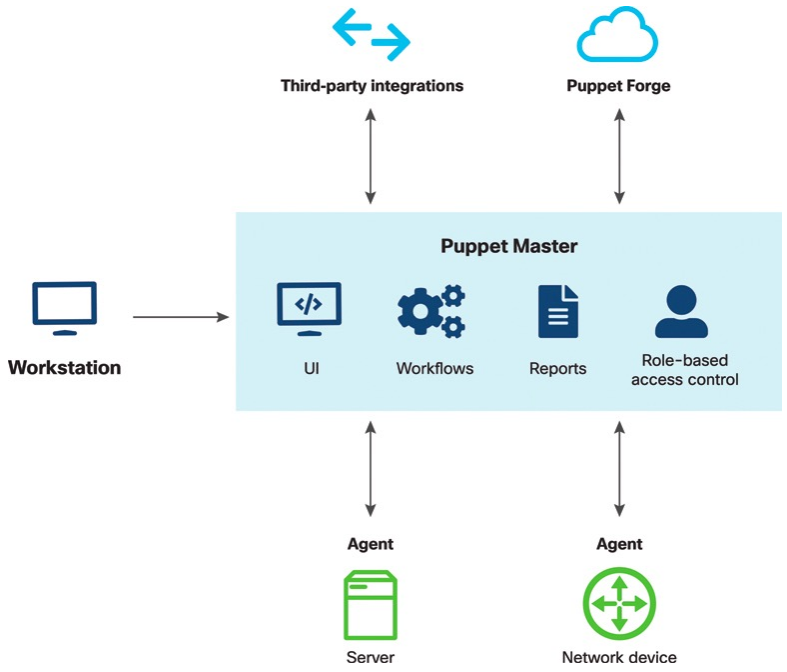
- Ansible
- Cisco NSO
- Salt Stack

Higher Order

- Terraform
- CloudFormation

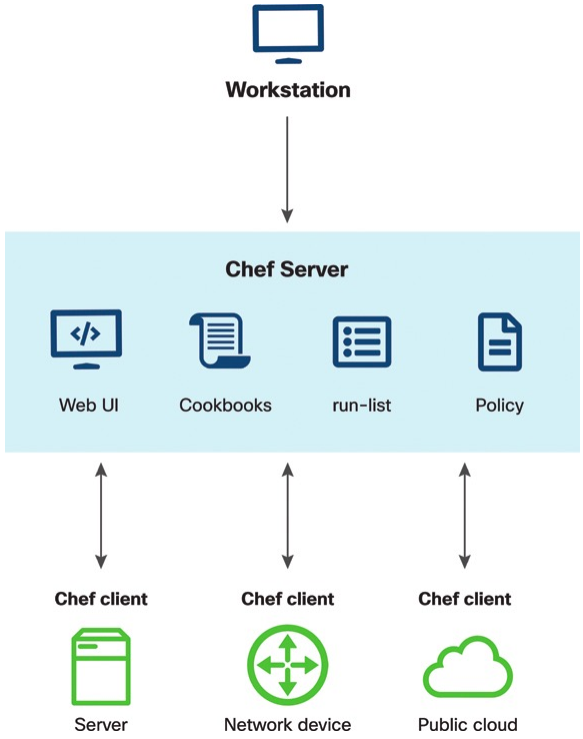
Puppet

- Configuration management platform written in Ruby
- Client-server architecture
- Polling-based communication
  - Agents check in every 30mins
- Declarative manifests written in Puppet DSL



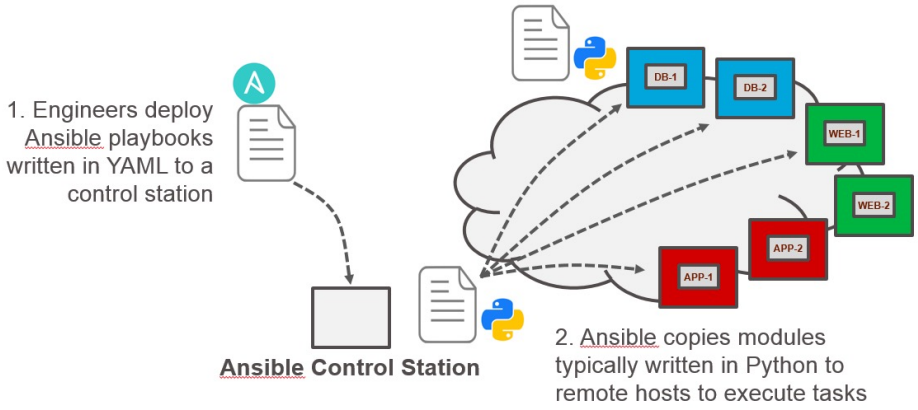
Chef

- Configuration management built on Ruby
- Client-server architecture
- Imperative model built around cookbooks and recipes
- Agent-based
- Polling communications



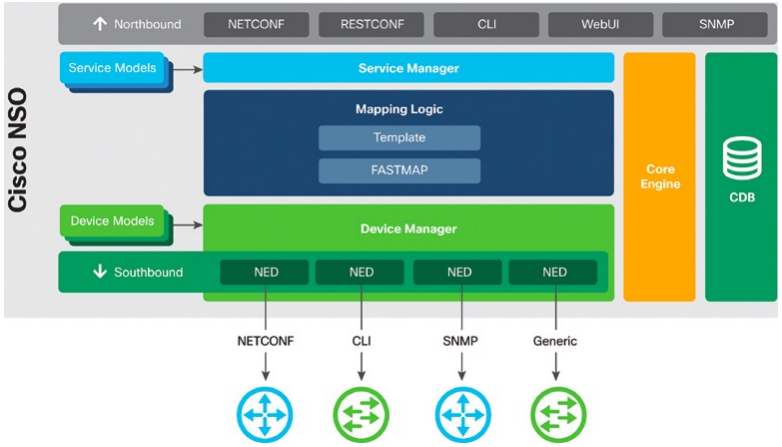
Ansible

- Written in Python
- Configures servers, applications, and networking
- Workflow described in YAML playbooks
- Agentless
  - SSH, NetConf, Rest API
- Huge list of modules and plugins



Cisco NSO

- Network Service Orchestration Platform
  - Service Manager
  - Device Manager
  - Mapping Logic
  - Configuration Database
- Uses NETCONF / YANG
- Management/Northbound API
  - REST, NETCONF, RESTCONF, JSON/RPC, CLI, Web UI, and SNMP
- Southbound control
  - NETCONF, CLI, SNMP, OpenFlow



Terraform

- Created by Hashicorp and very popular for managing DC and public cloud assets
- Compose and combine infrastructure resources to build and maintain a desired state
- Manages all resources through REST APIs
- Terraform uses built-in and plugin capabilities(providers) to enable control