



ANSIBLE

Unified Automation easily explained

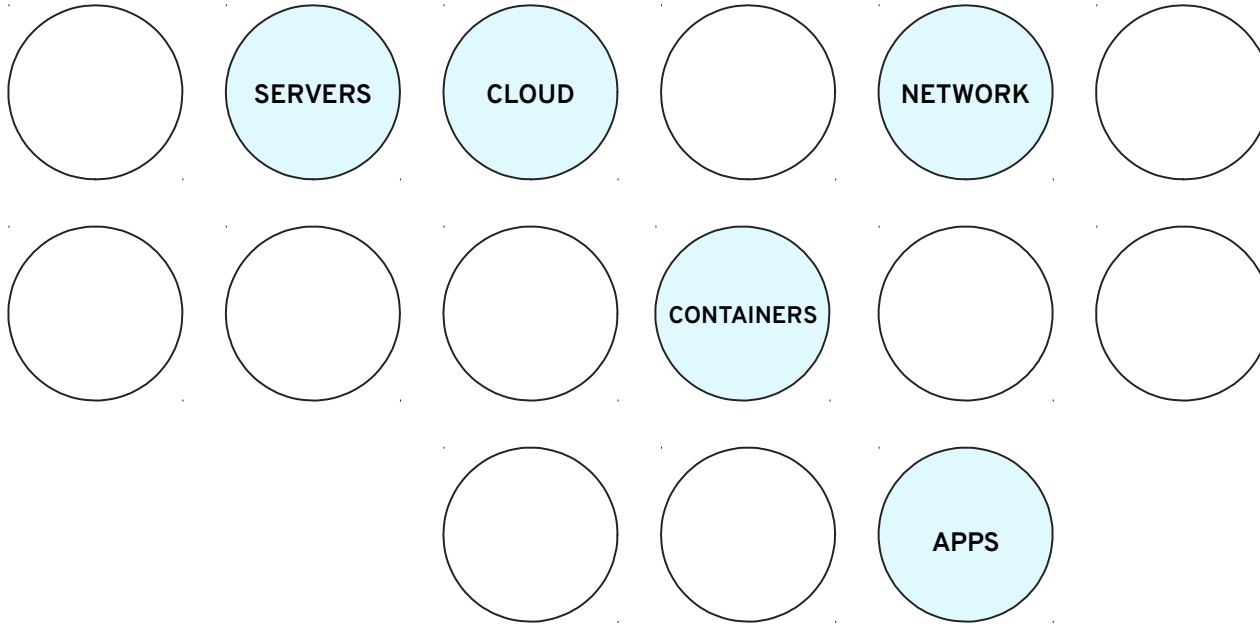
Ing. Günter Herold
Head of Cloud & Digital Transformation
November 2018



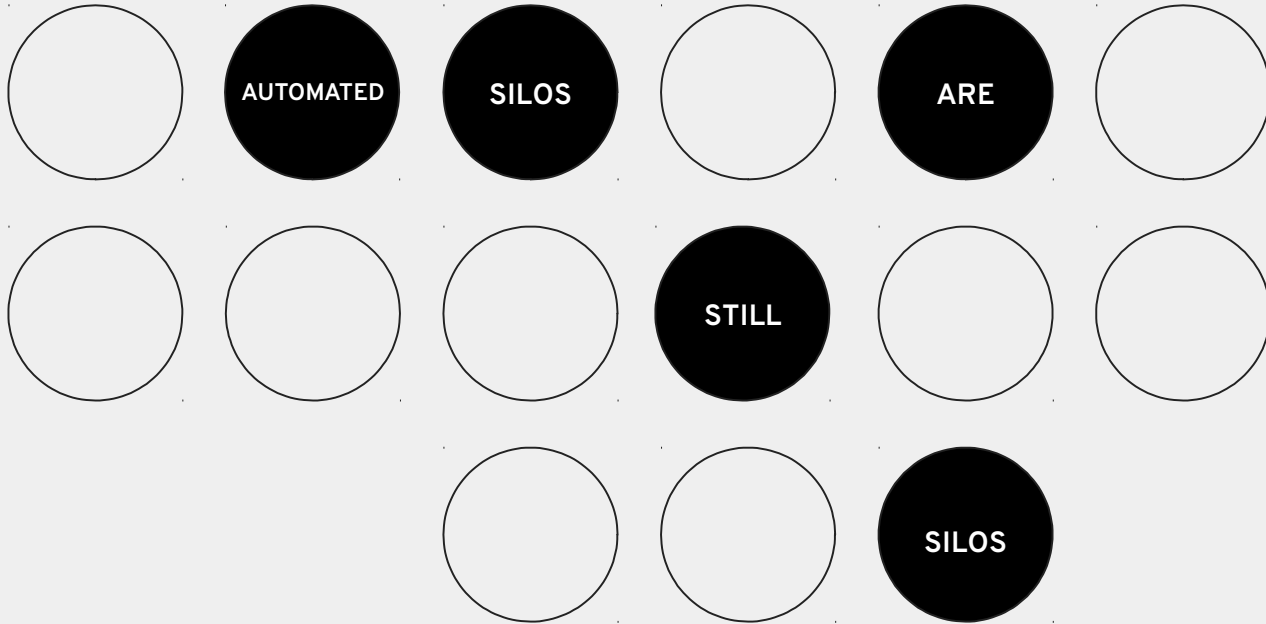
AUTOMATION FOR EVERYONE

Designed around the way people work
and the way people work together.

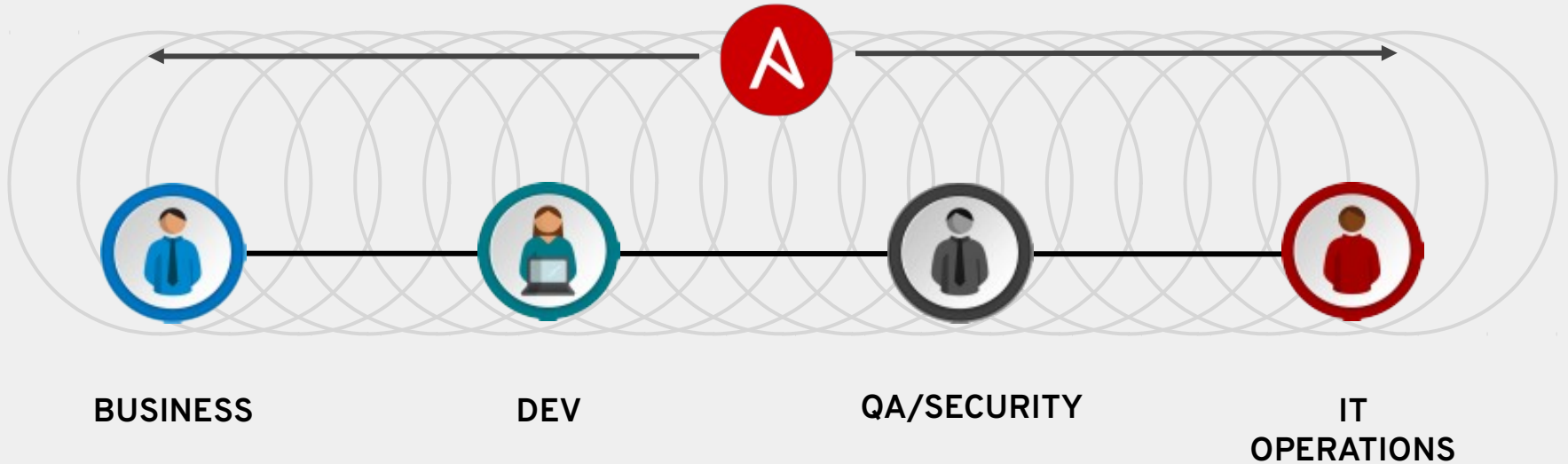
AUTOMATION IN ENTERPRISE IT TODAY



AUTOMATION IN ENTERPRISE IT TODAY



ANSIBLE IS THE UNIVERSAL LANGUAGE



ANSIBLE IS THE UNIVERSAL LANGUAGE

SIMPLE

Human readable
No coding skills needed
Tasks executed in order
Usable by every team

**GET PRODUCTIVE
QUICKLY**

POWERFUL

App deployment
Configuration management
Workflow orchestration
Network automation

**ORCHESTRATE
THE APP LIFECYCLE**

AGENTLESS

Agentless architecture
Uses OpenSSH & WinRM
No agents to exploit/update
Get started immediately

**MORE EFFICIENT
& MORE SECURE**



31,000+
Stars on GitHub

1900+
Ansible modules

500,000+
Downloads a month

ANSIBLE AUTOMATES TECHNOLOGIES YOU USE

Time to automate is measured in minutes

CLOUD

AWS
Azure
Digital Ocean
Google
OpenStack
Rackspace
+more

OPERATING SYSTEMS

RHEL and Linux
UNIX
Windows
+more

VIRT & CONTAINER

Docker
VMware
RHV
OpenStack
OpenShift
+more

STORAGE

NetApp
Red Hat Storage
Infinidat
+more

WINDOWS

ACLs
Files
Packages
IIS
Regedit
Shares
Services
Configs
Users
Domains
+more

NETWORK

Arista
A10
Cumulus
Bigswitch
Cisco
Cumulus
Dell
F5
Juniper
Palo Alto
OpenSwitch
+more

DEVOPS

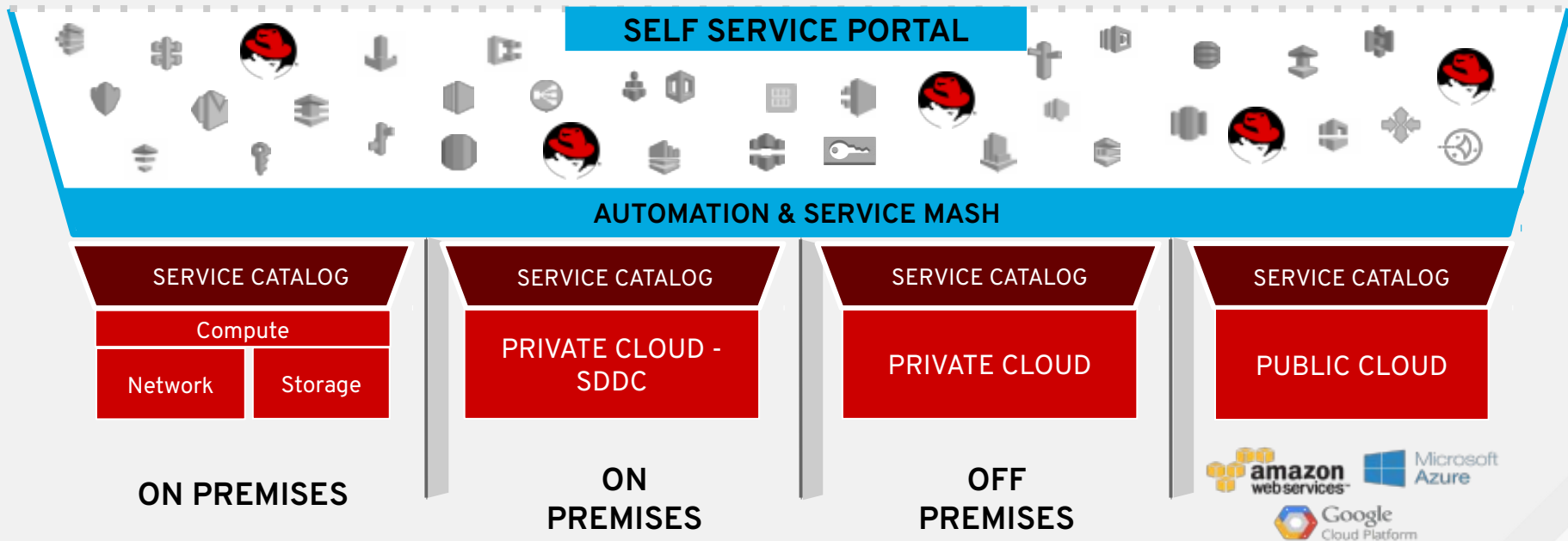
Jira
GitHub
Vagrant
Jenkins
Bamboo
Atlassian
Subversion
Slack
Hipchat
+more

MONITORING

Dynatrace
Airbrake
BigPanda
Datadog
LogicMonitor
Nagios
New Relic
PagerDuty
Sensu
StackDriver
Zabbix
+more

SELF-SERVICE FOR HYBRID SERVICES

VISION





RED HAT®
ANSIBLE®
Automation

RED HAT ANSIBLE TOWER

Scale + operationalize your automation

CONTROL

KNOWLEDGE

DELEGATION

RED HAT ANSIBLE ENGINE

Support for your Ansible automation

SIMPLE

POWERFUL

AGENTLESS

FUELED BY AN INNOVATIVE **OPEN SOURCE** COMMUNITY

DEPLOYMENT SCENARIO

RHEL IMG

W2k16 IMG

Internal Network

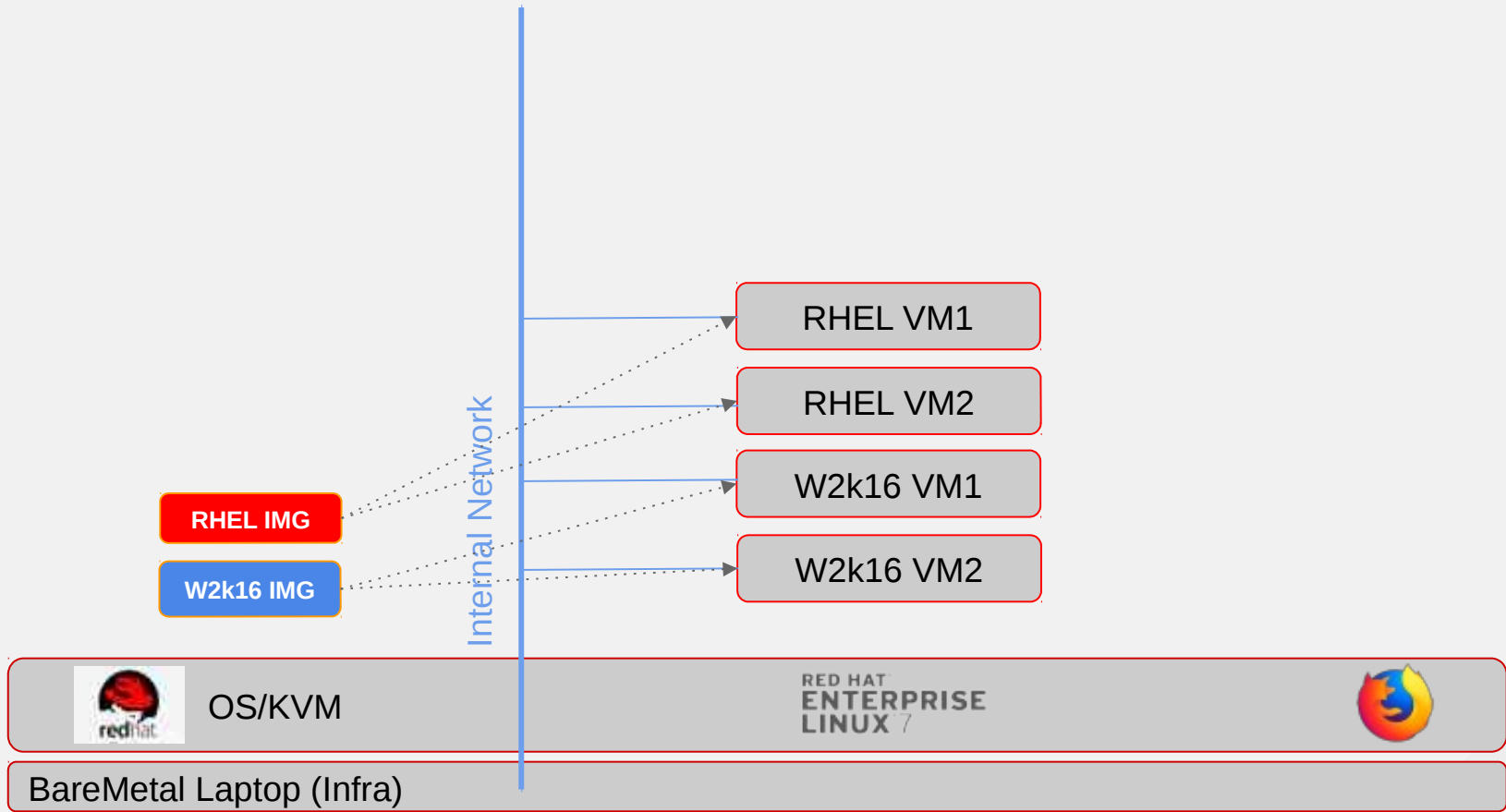


OS/KVM

RED HAT
ENTERPRISE
LINUX 7



BareMetal Laptop (Infra)



Create VMs:

Deploy 2 Windows VMs

Start, Pause 1 min.

Module:

command

virt

pause

```
40 lines (33 sloc) | 1.15 KB
Raw Blame History
1 ---
2 - name: manage libvirt guests
3   user: ansible
4   hosts: barelaptop
5   vars:
6     project_name: web
7     number_of_hosts: 2
8     os_type: win
9
10
11  tasks:
12    # not going to do xml_lookup just for a template, thank you.
13    # going with 'command' module and virt-clone
14    - name: create VM {{ os_type }}-{{ project_name }} items by cloning image
15      command: "virt-clone --original-win-image --name={{ os_type }}-{{ project_name }}{{ item }} --auto-clone"
16      with_sequence: count={{ number_of_hosts }}
17
18    - name: set description to {{ os_type }}-{{ project_name }} items
19      command: "virsh desc {{ os_type }}-{{ project_name }}{{ item }} --title {{ os_type }}-{{ project_name }}{{ item }}"
20      with_sequence: count={{ number_of_hosts }}
21
22  - name: start new VM
23    user: ansible
24    hosts: barelaptop
25    vars:
26      project_name: web
27      number_of_hosts: 2
28      os_type: win
29
30    tasks:
31      - name: start VM {{ os_type }}-{{ project_name }} items
32        virt:
33          name: "{{ os_type }}-{{ project_name }}{{ item }}"
34          state: running
35        with_sequence: count={{ number_of_hosts }}
36
37      - name: allow DHCP/DNS to register
38        pause:
39          minutes: 1
```

Create VMs:

Ansible Tower GUI

The screenshot displays the Ansible Tower web interface. The browser address bar shows the URL `https://tower/#/jobs/playbook/6967/job_search?page_size:20;order_by:-finished,not_launched,launch_type:sync`. The page title is "TOWER". The main content area shows a job execution summary for "696 - SuperCorp Create RHEL VMs".

Job Details:

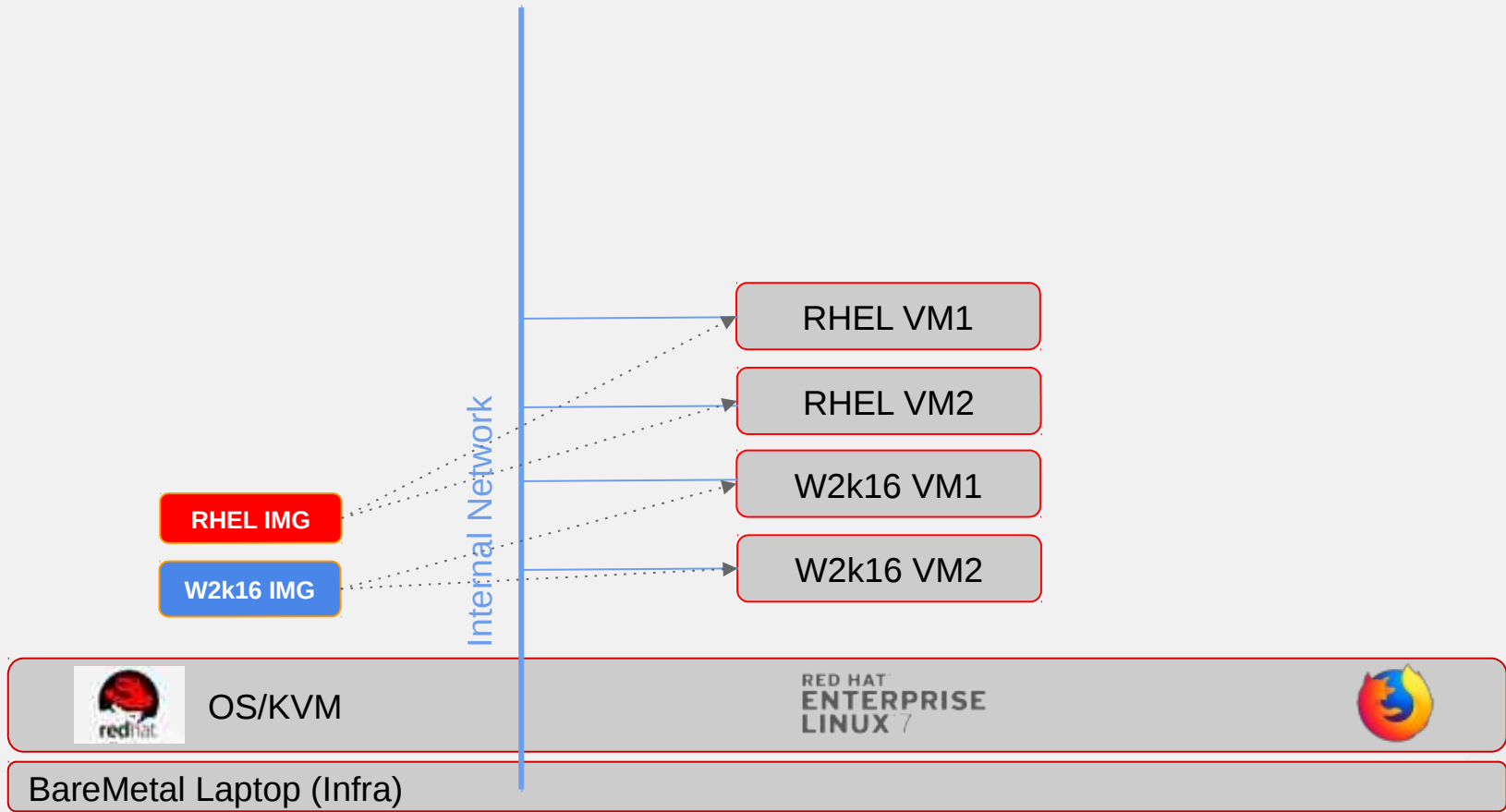
- STATUS: Successful
- STARTED: 10/24/2018 12:21:28 PM
- FINISHED: 10/24/2018 12:22:56 PM
- JOB TEMPLATE: SuperCorp Create RHEL VMs
- JOB TYPE: Run
- LAUNCHED BY: admin
- INVENTORY: SuperCorp Dynamic Inventory
- PROJECT: Automation in AT with SuperCorp and SCM
- REVISION: f2edd56
- PLAYBOOK: rhel_vmcreate.yml
- CREDENTIAL: SuperCorp Server Credentials
- INSTANCE GROUP: tower

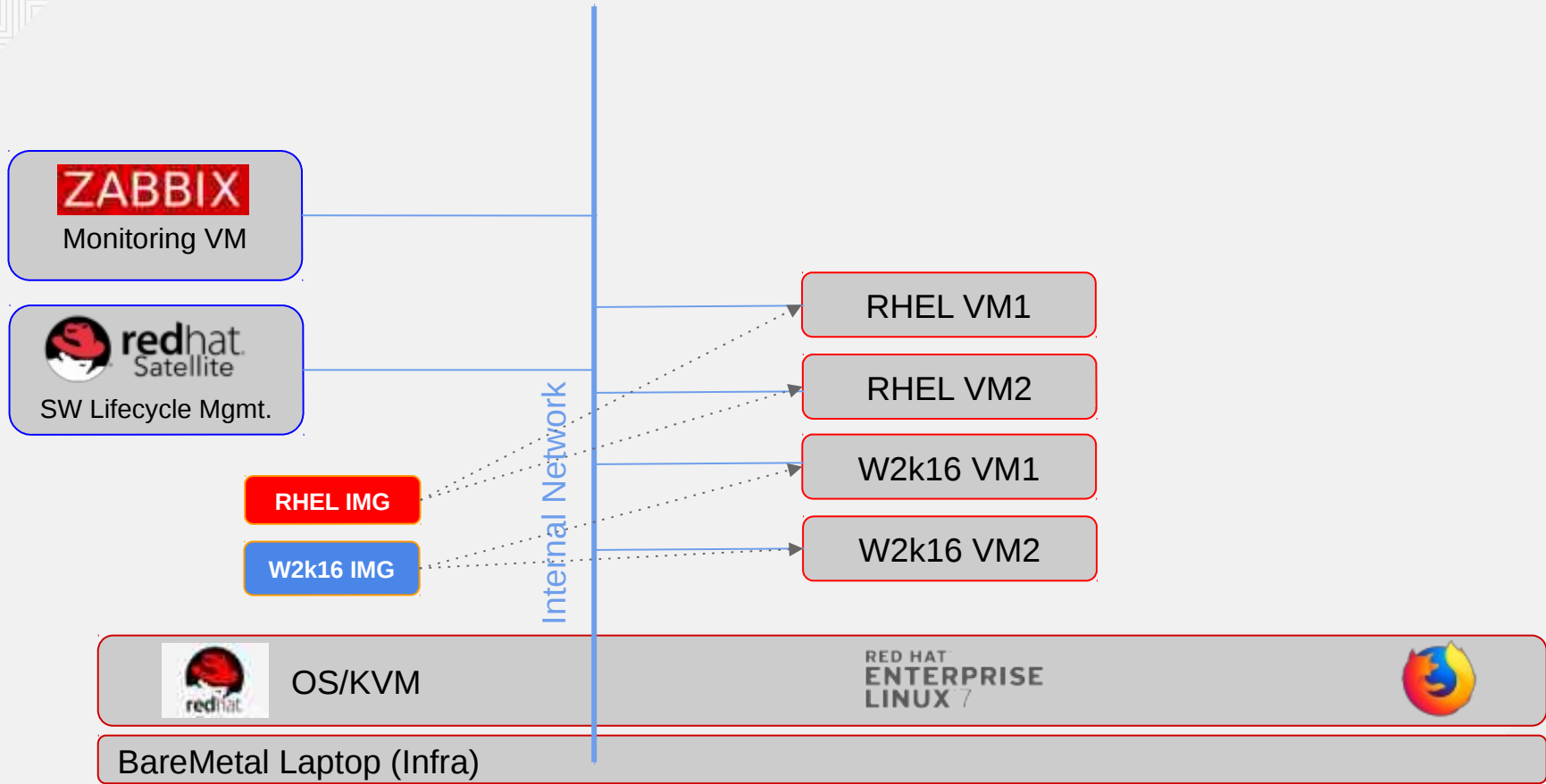
Extra Variables:

```
1 number_of_hosts: 1
2 project_name: infra
3
```

Job Output:

```
5
6 TASK [Gathering Facts] *****
7 ok: [gateway]
8
9 TASK [create VM rhel-infra items by cloning] *****
10 changed: [gateway] => (item=1)
11
12 TASK [sysprep VM rhel-infra items] *****
13 changed: [gateway] => (item=1)
14
15 PLAY [start new guests] *****
16
17 TASK [Gathering Facts] *****
18 ok: [gateway]
19
20 TASK [start VM "rhel-infra items"] *****
21 changed: [gateway] => (item=1)
22
23 TASK [allow DHCP/DNS to register] *****
24 Pausing for 60 seconds
25 (ctrl+C then 'C' = continue early, ctrl+C t
26 ok: [gateway]
27
```





ZABBIX Setup

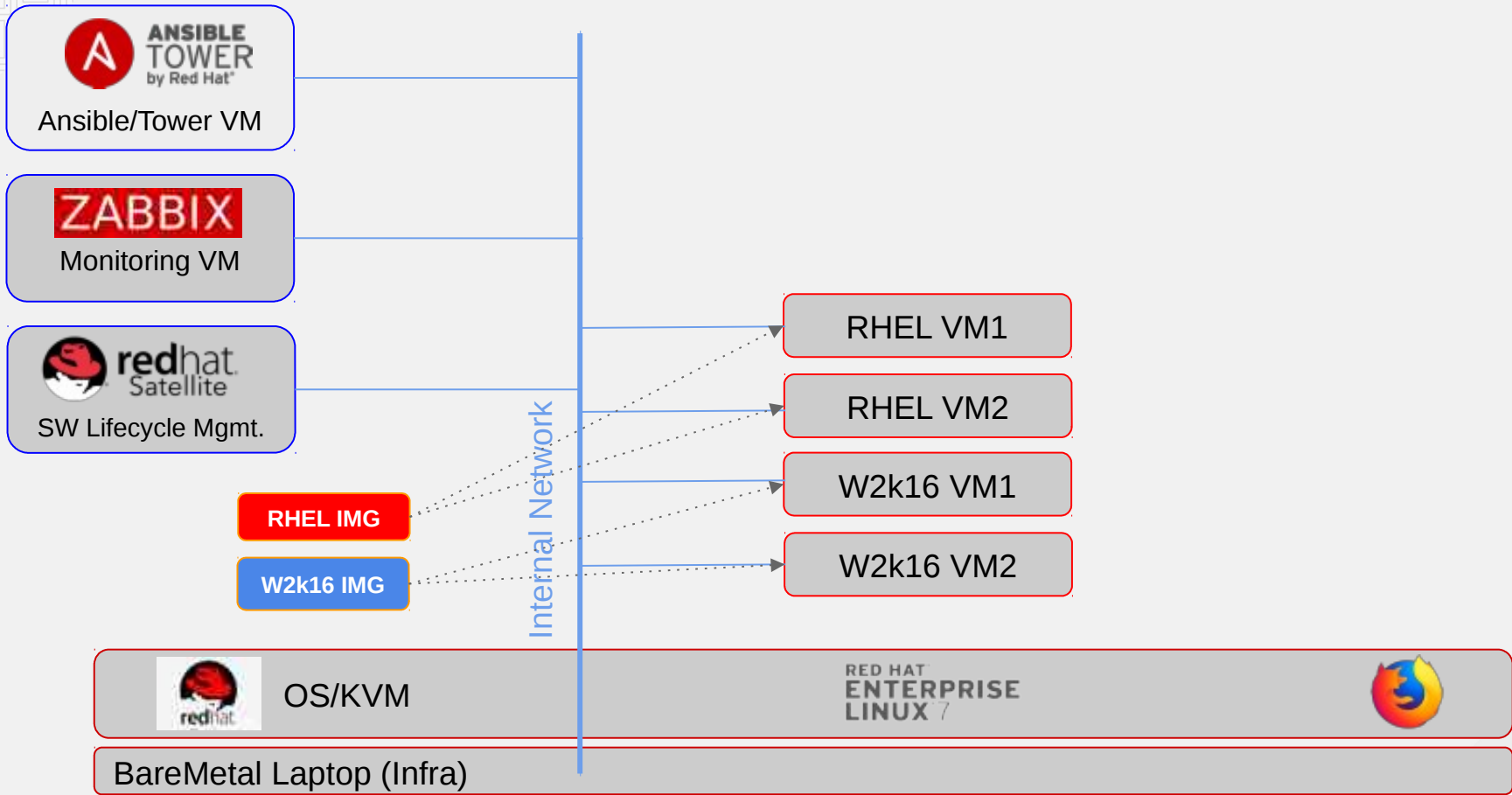
Module:

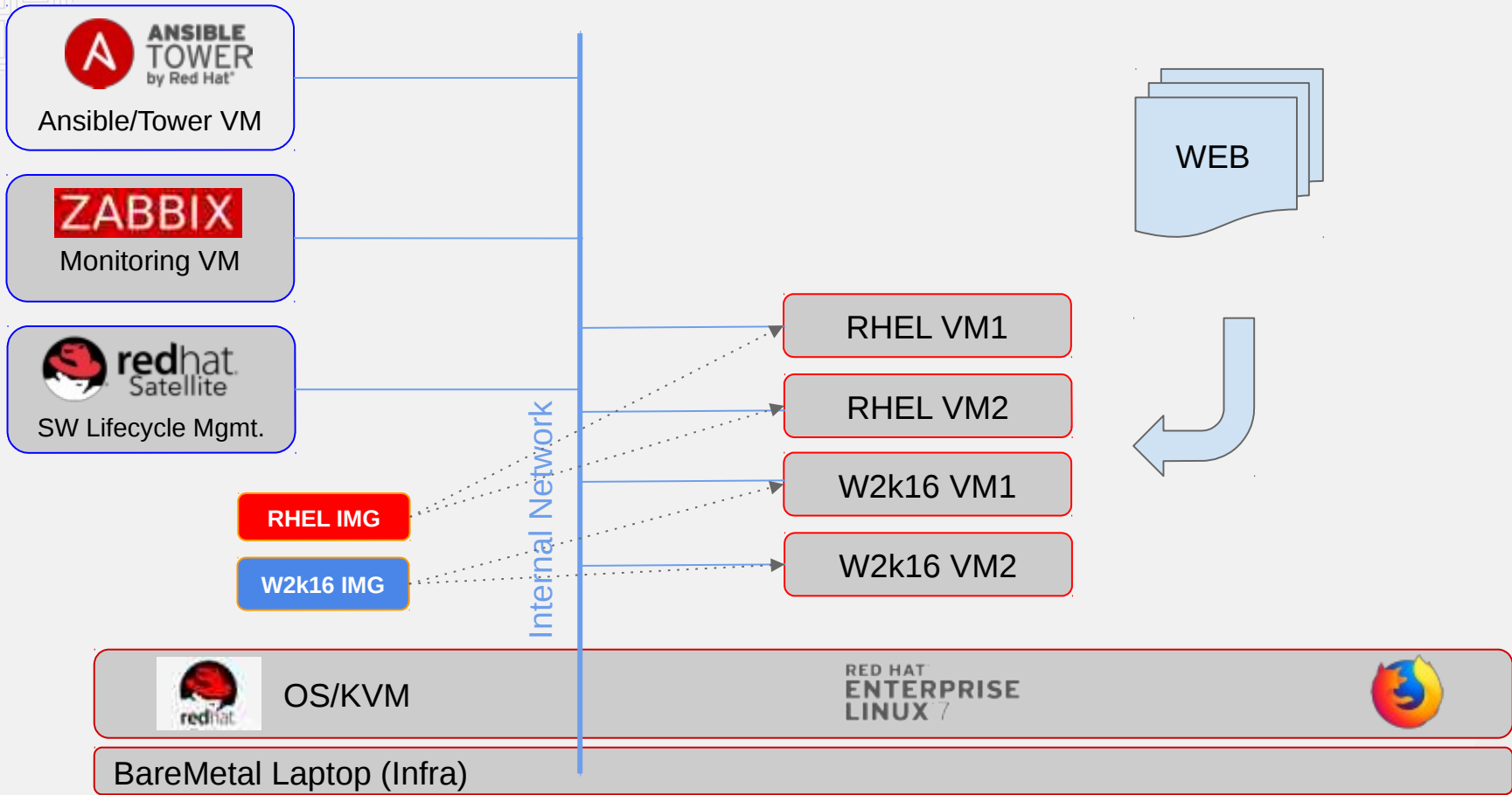
yum

systemd

zabbix_host

```
47 lines (41 sloc) | 1.2 KB
Raw Blame History
1 ---
2 - name: Setup Zabbix client
3   users: ansible
4   hosts: rhel_vms
5
6   tasks:
7     - name: Install Zabbix agent
8       yum:
9         name: zabbix30-agent
10        state: latest
11
12    - name: make sure zabbix client service is enabled
13      systemd:
14        name: zabbix-agent
15        enabled: yes
16
17    - name: Create a new host or update an existing host's info
18      local_action:
19        module: zabbix_host
20        server_url: http://zabbix/zabbix
21        login_user: Admin
22        login_password: zabbix
23        host_name: "{{ inventory_hostname }}"
24        visible_name: "{{ inventory_hostname }}"
25        description: This is the {{ inventory_hostname }} description
26        inventory_mode: manual
27        host_groups:
28          - RHEL Servers
29        link_templates:
30          - Template App HTTP Service
31          - Template App SSH Service
32          - Template Module ICMP Ping
33          - Template OS Linux
34        status: enabled
35        state: present
36        interfaces:
37          - type: agent
38            main: 1
39            dns: "{{ inventory_hostname }}"
40            ip: "{{ ansible_eth0.ipv4.address }}"
41
42    - name: make sure zabbix client is running
43      systemd:
44        name: zabbix-agent
45        state: started
46
```





Enable Webserver

Module:

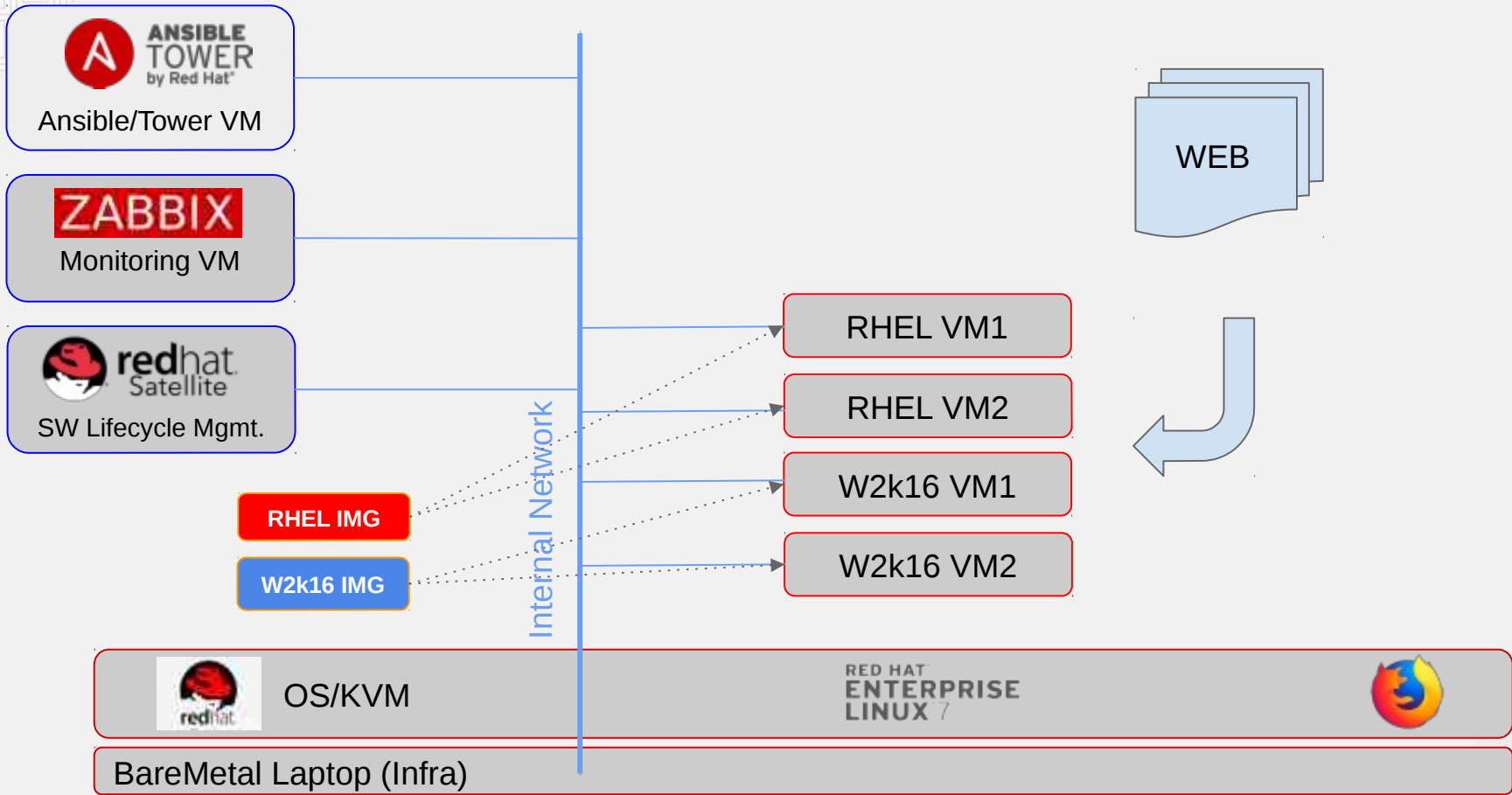
win_feature

register

win_reboot

17 lines (14 sloc) | 399 Bytes

```
1 ---
2 - name: webserver setup
3   hosts: win_vms
4
5   tasks:
6     - name: Install IIS Web-Server with sub features and management tools
7       win_feature:
8         name: Web-Server
9         state: present
10        include_sub_features: yes
11        include_management_tools: yes
12        register: win_feature
13
14     - name: reboot if installing Web-Server feature requires it
15       win_reboot:
16         when: win_feature.reboot_required
```



ANSIBLE NETWORK SCENARIO



NETWORK AUTOMATION

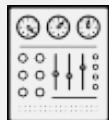
CONFIGURE, VALIDATE, & ENSURE CONTINUOUS COMPLIANCE FOR PHYSICAL NETWORK DEVICES



BUILD

with Red Hat Ansible Engine

- Get automating quickly.
- Integrate multivendor configurations.
- Ideal for both brownfield and greenfield.



MANAGE

with Red Hat Ansible Engine

- Methodically track configuration drift.
- Make changes across any set of network devices.
- Validate that changes were successful.



SCALE

with Red Hat Ansible Tower

- Ensure ongoing steady state on a schedule.
- Use role-based access controls with specific teams.
- Integrate with RESTful API.

Automate discrete tasks

Automate business processes

Orchestrate & operationalize automation



WHAT IS THE PRIMARY METHOD OF MAKING NETWORK CHANGES IN YOUR ENVIRONMENT?

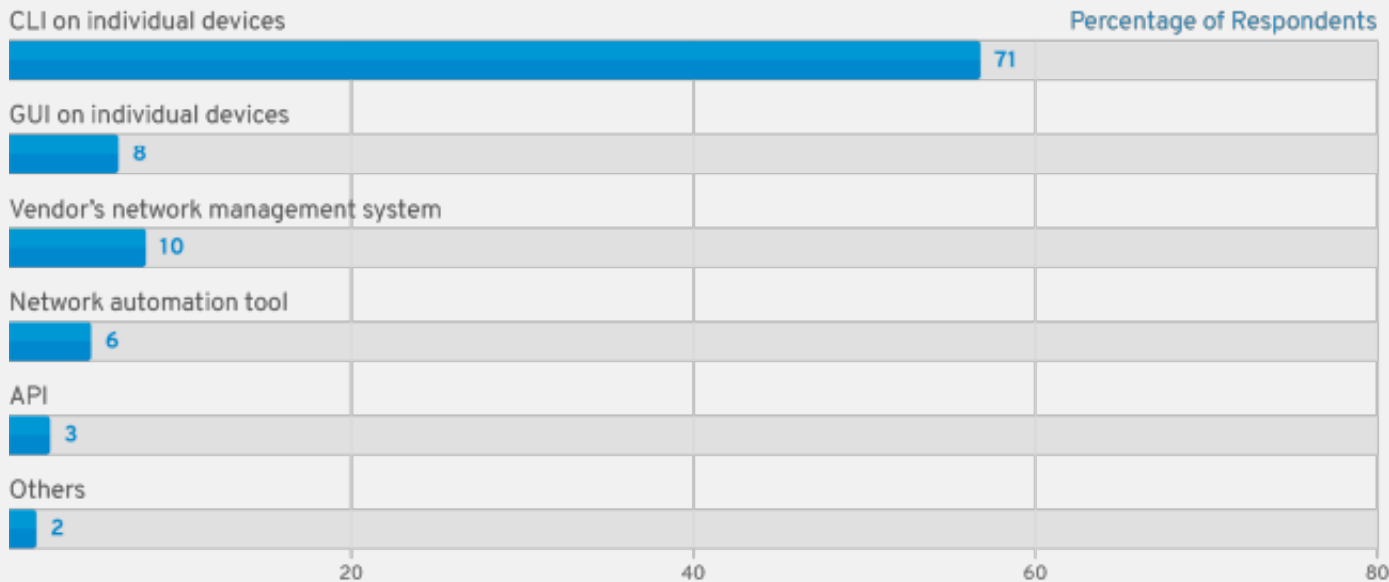


Figure 1
Primary Method for Making Network Changes

Source: Gartner, *Look Beyond Network Vendors for Network Innovation*, Andrew Lerner, 23 January 2018, ID: G00349636.



WHY RED HAT

FOR NETWORK AUTOMATION



Tactics and technologies like SDN, host networking, virtual networking, control plane, or data plane are important, but they are still only implementations.

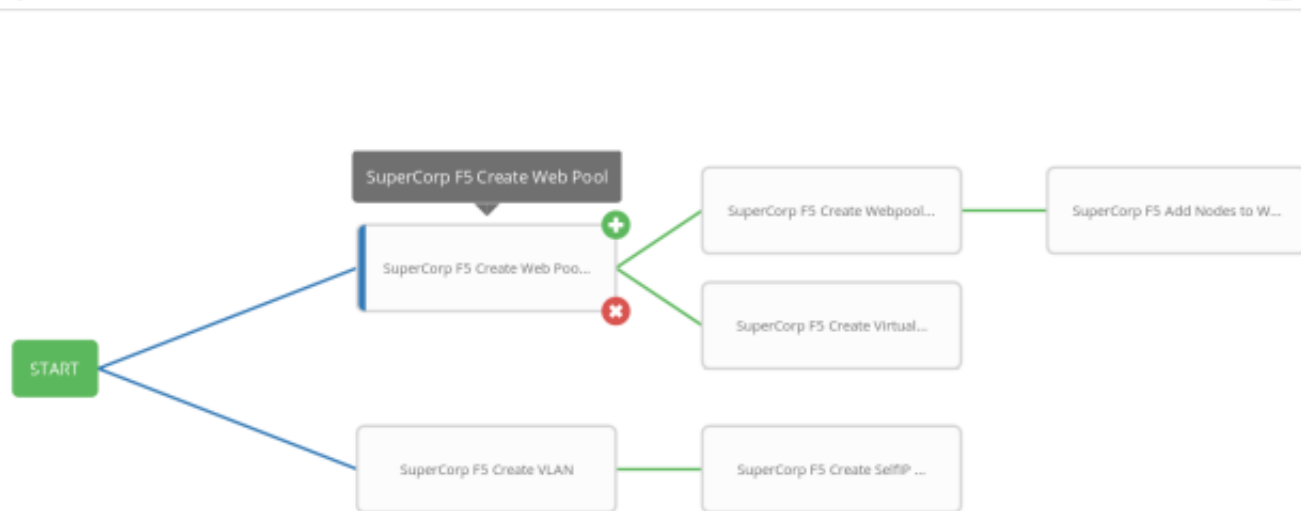


Think strategically. Enable your customers to more easily **operate**, **augment**, **simplify**, **scale**, and ultimately **consume** those network resources.

WORKFLOW VISUALIZER | SuperCorp FS LoadBalancer for webservers

Q

TOTAL TEMPLATES 13



SUPERCORP FS CREATE WEB POOL

JOBS

PROJECT SYNC

INVENTORY SYNC

SEARCH



KEY

NAME *

 Demo Job Template

INFO

 S IT Win Test Job Template

INFO

 SuperCorp app: RHEL webserver setup

INFO

 SuperCorp app: Windows webserver setup

INFO

 SuperCorp Create RHEL VMS

INFO

 1 2 3 4 5

PAGE 1 OF 5

ITEMS 1 - 5 OF 23

* RUN

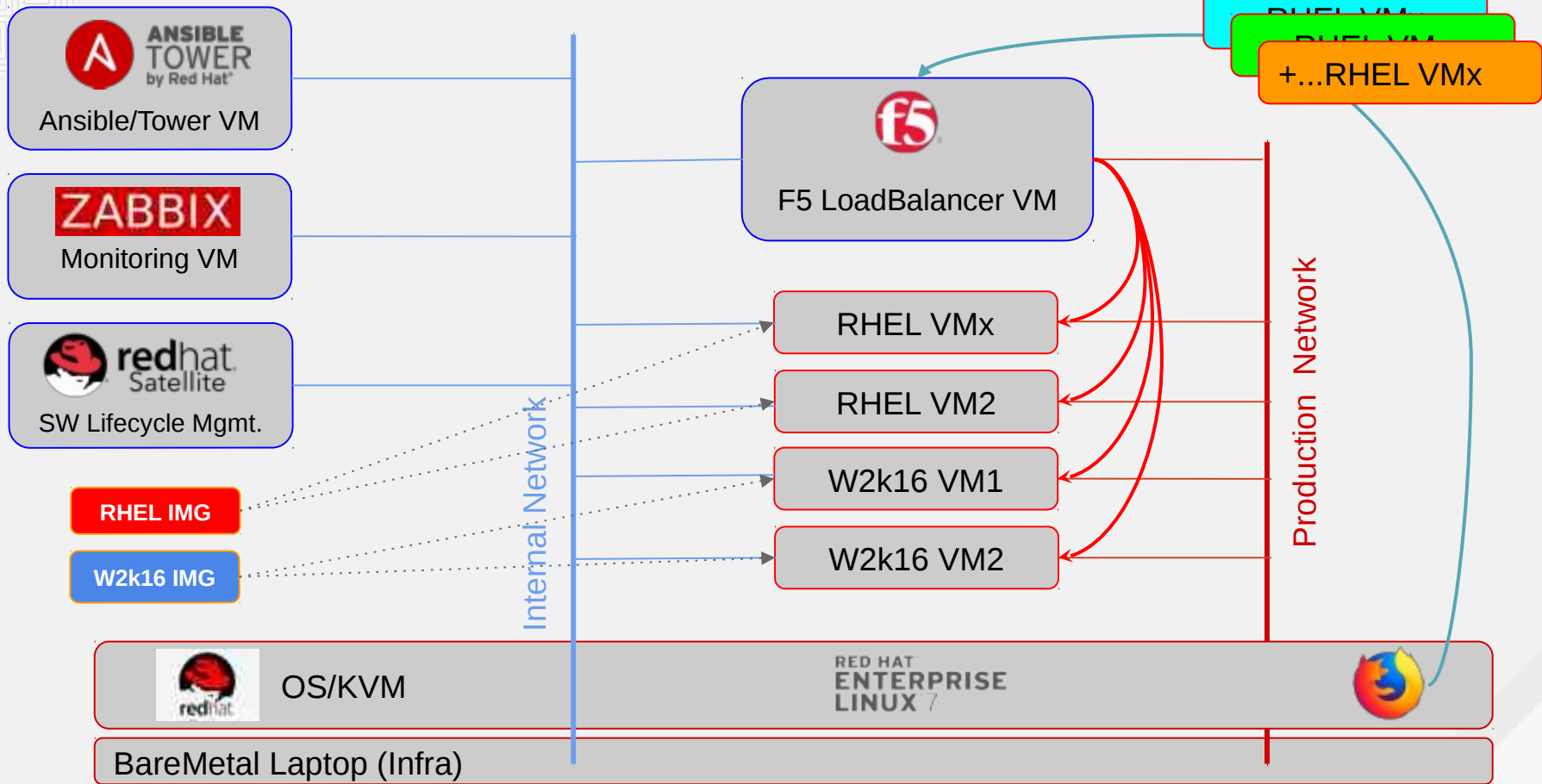
Always

CANCEL

SELECT

CLOSE

SAVE



PROMPT

SURVEY

PREVIEW

* ABTEILUNG

* KOSTENSTELLE

CPU

MEMORY

STORAGE

CANCEL

NEXT

+...RHEL VMx

File Edit View History Bookmarks Tools Help

Managing Hosts - Re... Content Management... https://sat... Zaaaacc12 Ansible Tower | DAS... +

https://tower/#!/home 110% what is hammer cli

Red Hat Support I.T. Toolbox Charlessstuff ISA New Request FAQ: SA DACH | Mojo Plan A RHVH Getting started f... OneStop

TOWER admin

13 HOSTS

1 FAILED HOSTS

3 INVENTORIES

0 INVENTORY SYNC FAILURES

3 PROJECTS

0 PROJECT SYNC FAILURES

JOB STATUS

PERIOD: **PAST MONTH** JOB TYPE: **ALL** VIEW: **ALL**

RECENTLY USED TEMPLATES

[VIEW ALL](#)

NAME	ACTIVITY	ACTIONS
SuperCorp F5 cleanup	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	
SuperCorp F5 drop Nodes	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	
SuperCorp F5 drop VLAN	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	
SuperCorp F5 drop Pool	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	
SuperCorp F5 drop SelfIP	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	

RECENT JOB RUNS

[VIEW ALL](#)

NAME	TIME
● SuperCorp F5 cleanup	10/22/2018 1:57:33 PM
● SuperCorp F5 drop Nodes	10/22/2018 1:57:33 PM
● SuperCorp F5 drop VLAN	10/22/2018 1:57:23 PM
● SuperCorp F5 drop Pool	10/22/2018 1:57:17 PM
● SuperCorp F5 drop SelfIP	10/22/2018 1:57:10 PM

FORRESTER®

The Total Economic Impact Of Red Hat Ansible Tower – Webinar for Spark

Sam Conway, TEI Consultant

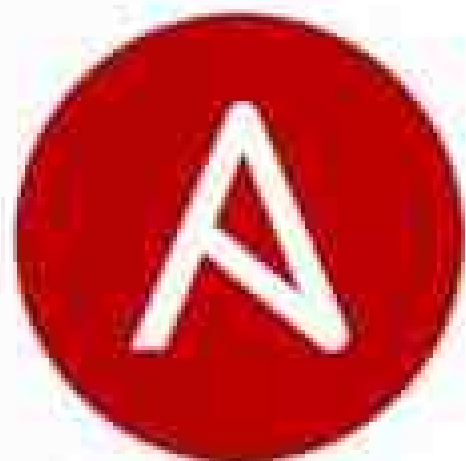
September 12, 2018



A woman with blonde hair tied back is sitting in the driver's seat of a car, looking down at a newspaper she is holding. The car's interior is visible, including the steering wheel, dashboard, and center console. Through the windshield, a road with other cars is visible. The text 'AUTOMATION FOR EVERYONE' is overlaid in large white letters across the center of the image.

AUTOMATION FOR EVERYONE

Designed around the way people work
and the way people work together.



ANSIBLE
TOWER
by Red Hat®



THANK YOU



plus.google.com/+RedHat



facebook.com/redhatinc



linkedin.com/company/red-hat



twitter.com/RedHat



youtube.com/user/RedHatVideos