

Infraestructura como código para entornos DevOps y Software-defined datacenter

Pablo Ráez

Category Manager

Hewlett Packard Enterprise

OPEN
24/7

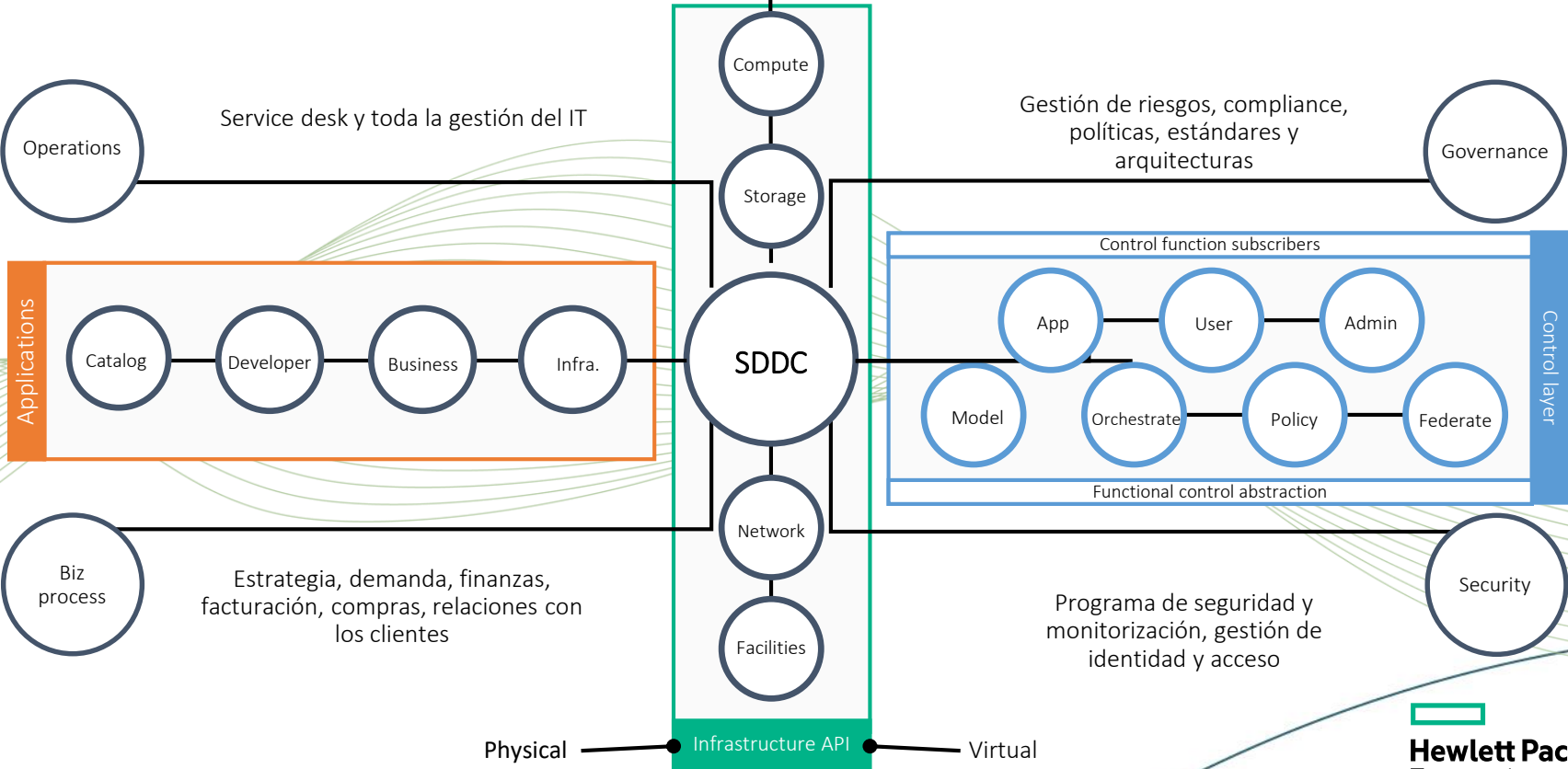
¿Quieres
entrar?

Hewlett Packard Enterprise

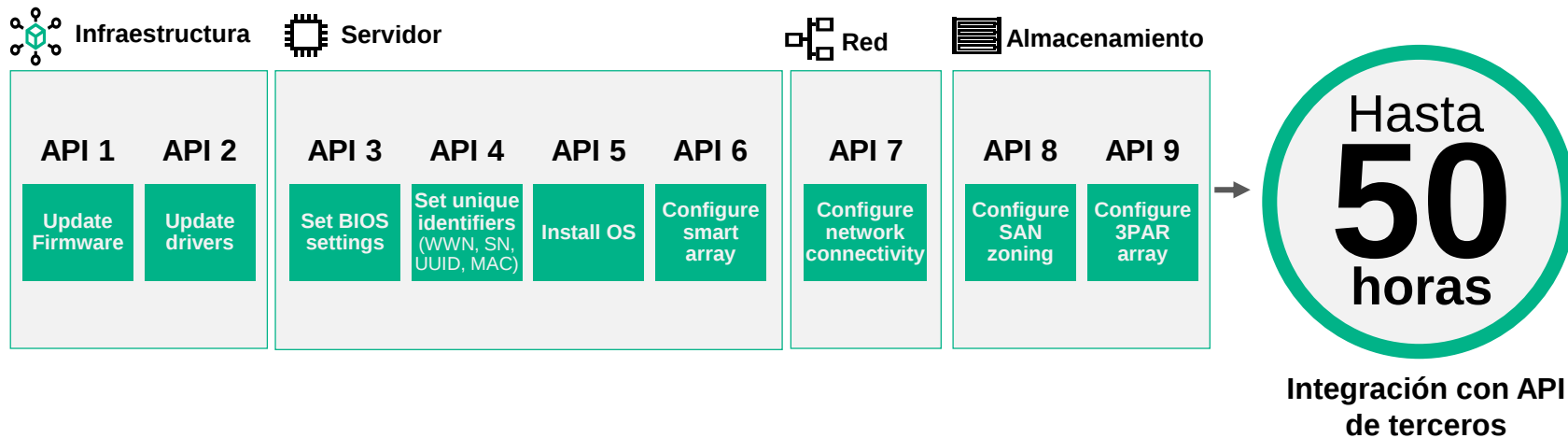


```
24 <meta charset="utf-8" />
25 <meta name="viewport" content="width=device-width, initial-scale=1" />
26 <meta name="description" content="WordPress" />
27 <meta name="author" content="WordPress" />
28 <meta name="generator" content="WordPress" />
29 <meta name="language_attributes" />
30 <meta name="bloginfo" charset="utf-8" />
31 <meta name="profile" href="http://gmpg.org/xfn/11" />
32 <meta name="pingback" href="http://gmpg.org/xfn/11" />
33 <meta name="fruitful_get_favicon" />
34 <meta name="fruitful_get_favicon" />
35 <meta name="fruitful_get_favicon" />
```


Software-defined Datacenter



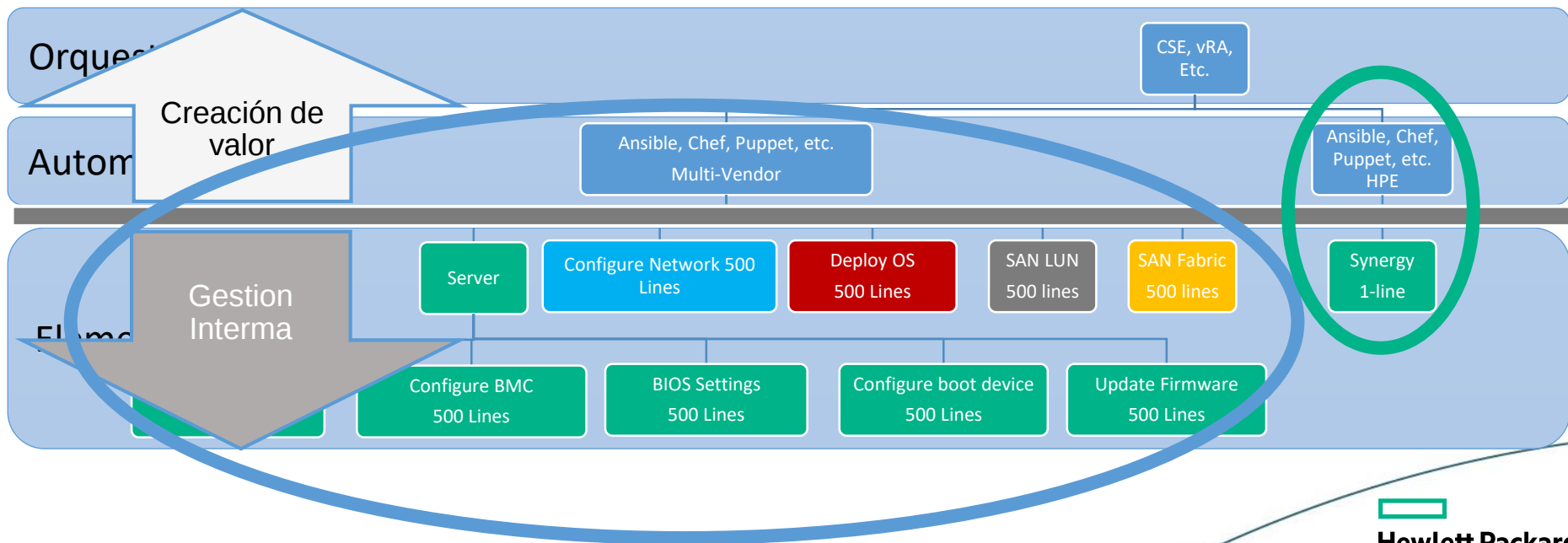
Transformar la infraestructura con líneas de código



¹ Clutch [Cost to Build a Mobile App: A Survey](#)

Despliegue aplicaciones más rápidamente

Dedique menos tiempo en gestiones internas, más tiempo desarrollando su aplicación



Visión de HPE: “Composable Infrastructure”

Ayudar al TI de los Clientes a afrontar los retos de la nueva era digital



Ejecutar cualquier carga, almacenar todo



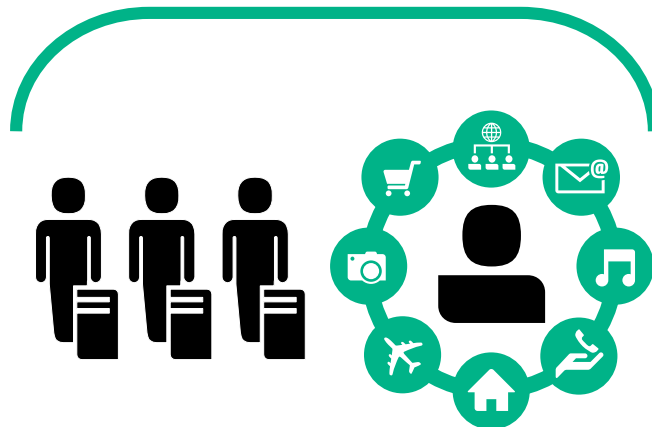
Entregar aplicaciones y servicios de forma espectacularmente más rápida



Simplificar las operaciones e integración del ciclo de vida



Agiliza la obtención de valor, aumenta la productividad y el control



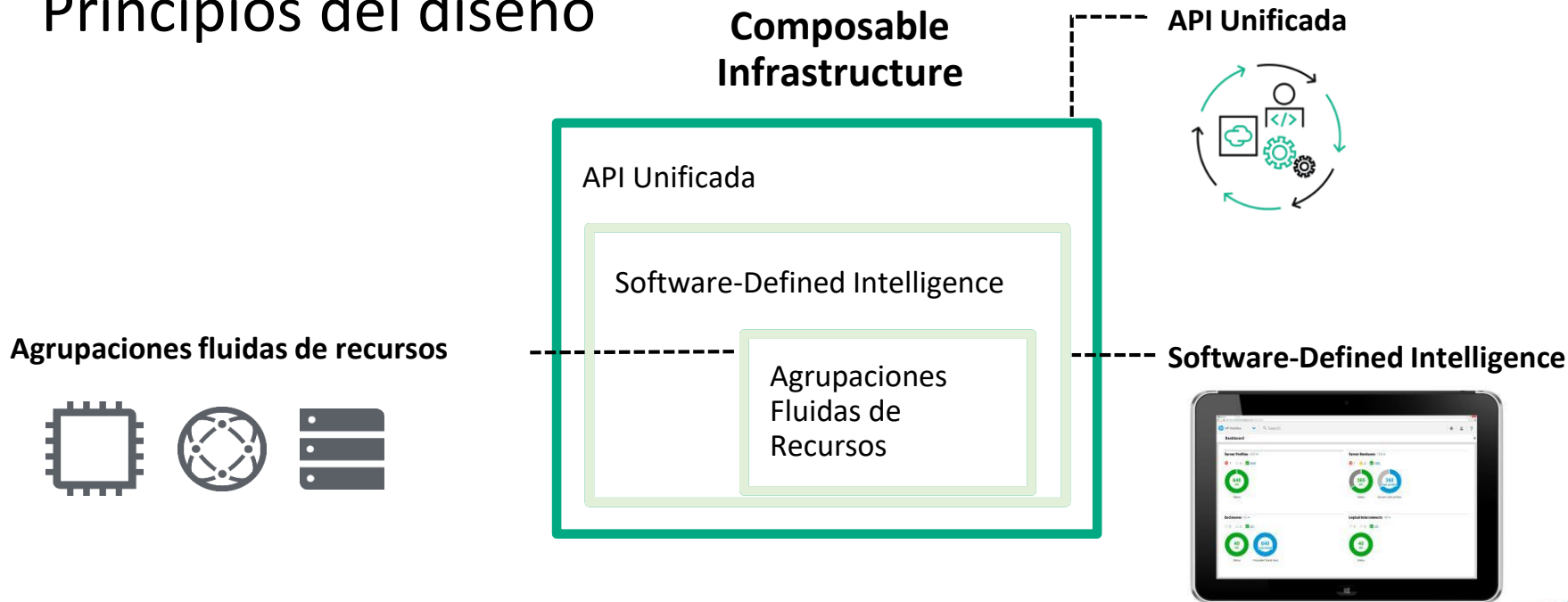
El Software define su infraestructura - Asesorar | Transformar | Integrar

Su infraestructura configurada con una sola línea de código



Composable Infrastructure

Principios del diseño



Transformar la infraestructura con líneas de código

API Unificada

Reduce el tiempo de
aprovisionamiento de la
infraestructura de



horas

a

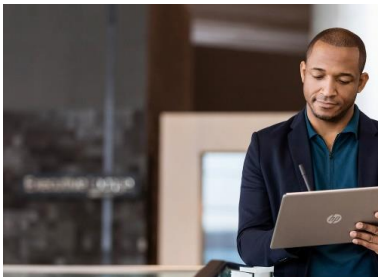


minutos



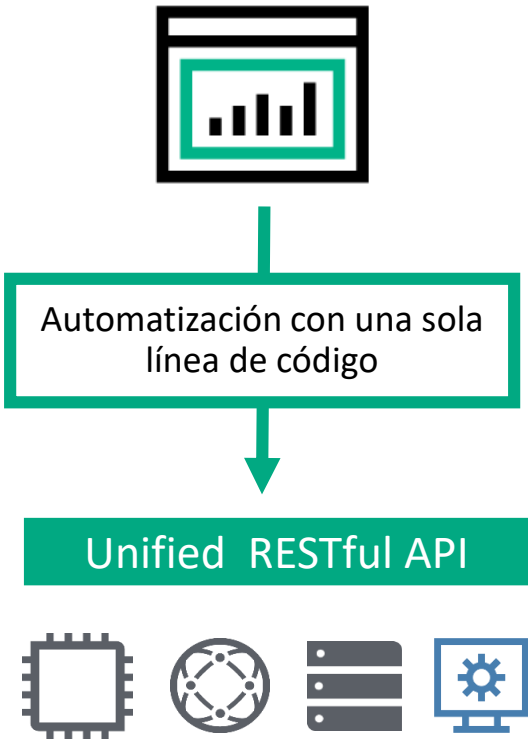
```
New-HPOVProfile -name $name -baseline $baseline -sanstorage $san -server $server
```



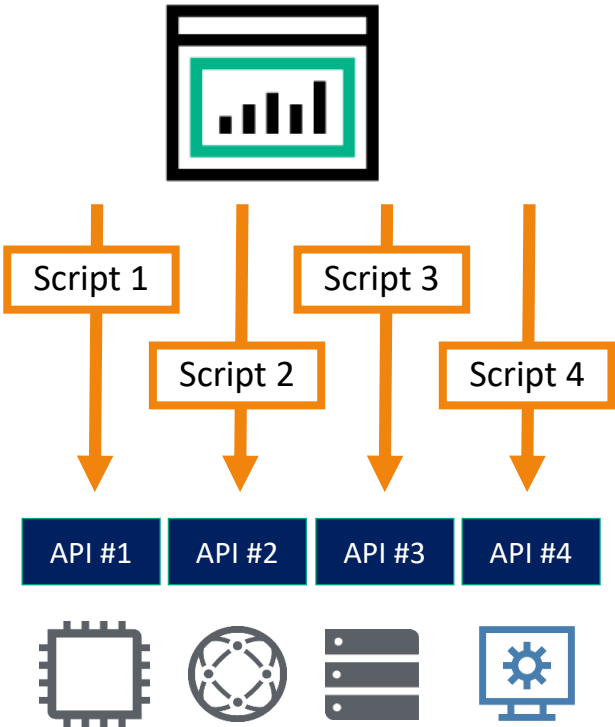


“Puedo solicitar los recursos necesarios sin ser un experto en infraestructura”

Con HPE IaC

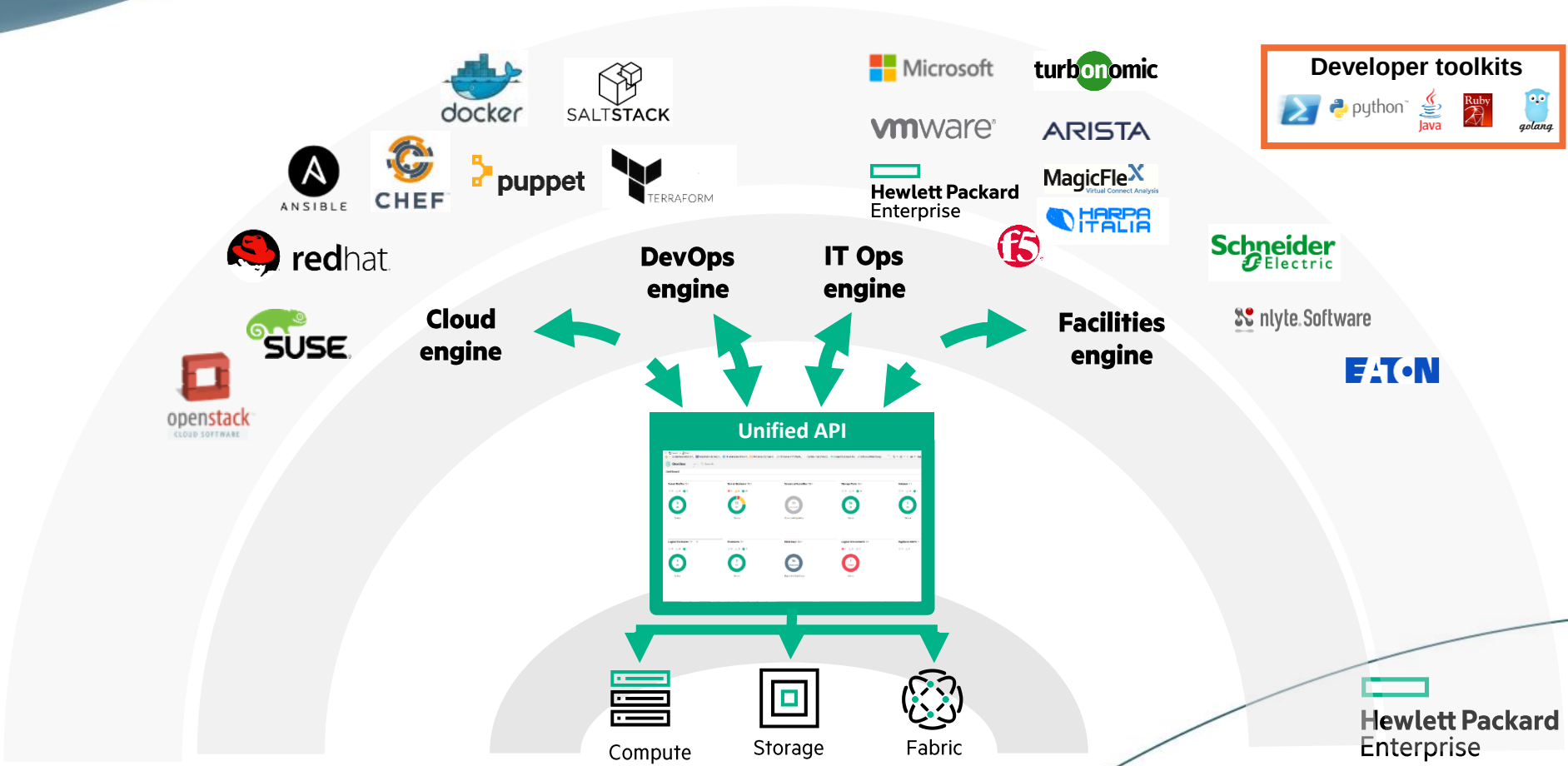


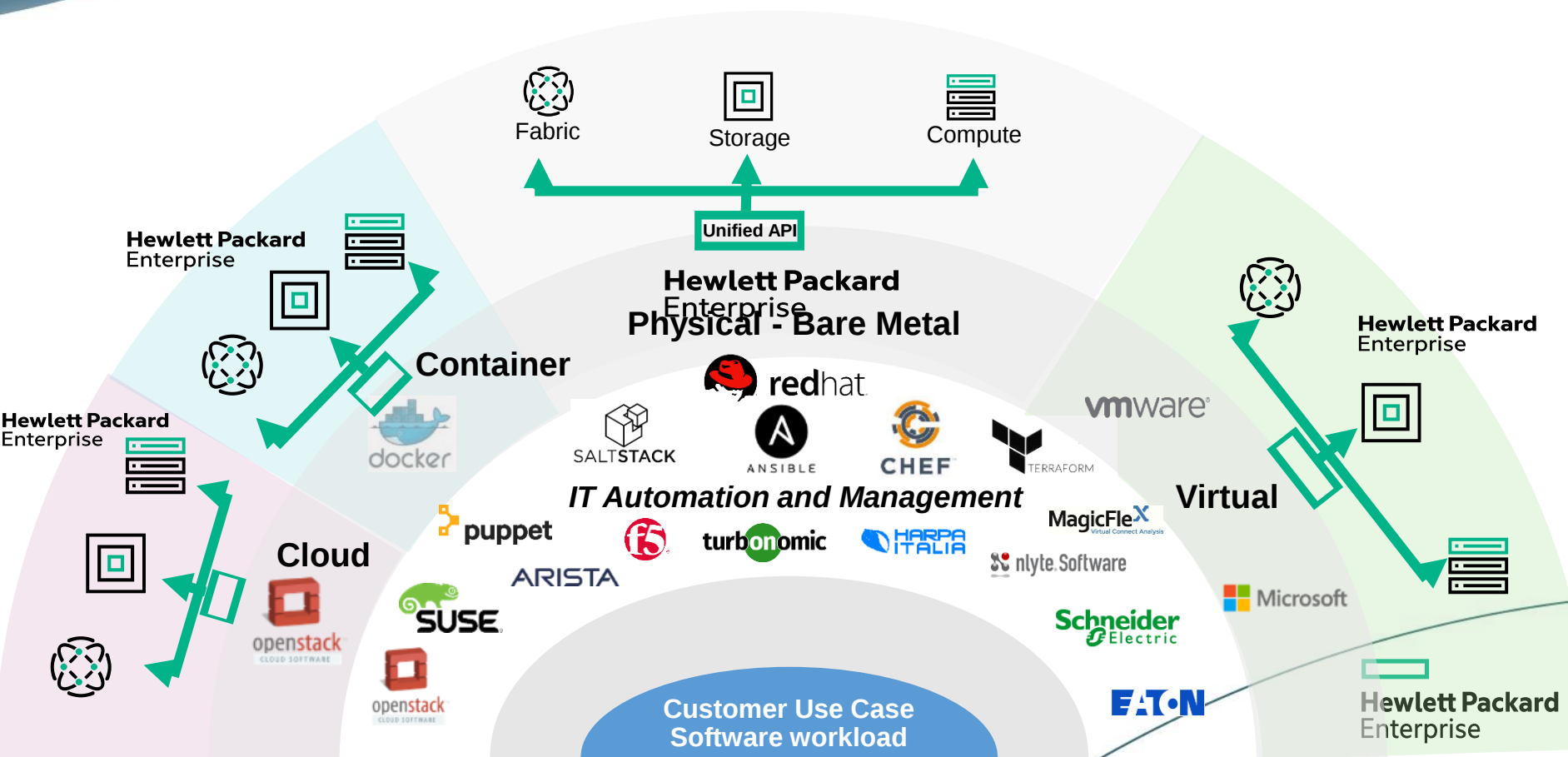
Sin IaC





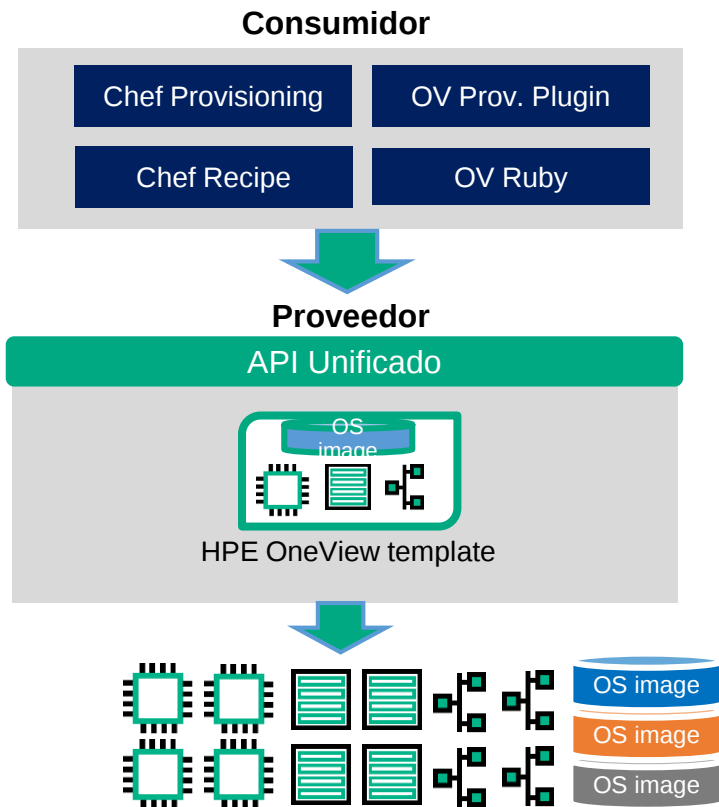
¿Quién está ya en el Ecosistema?





Un ejemplo de integración: CHEF

Cómo desplegar una aplicación web en minutos



1. Definimos un **template** de infraestructura (OV)

2. Despliegue automático de infraestructura (C+OV)

3. Provision de políticas, integración y Aplicación (C)

4. Y...a otra cosa!

Infraestructura Componible en acción con Chef

Acelerar el time-to-value

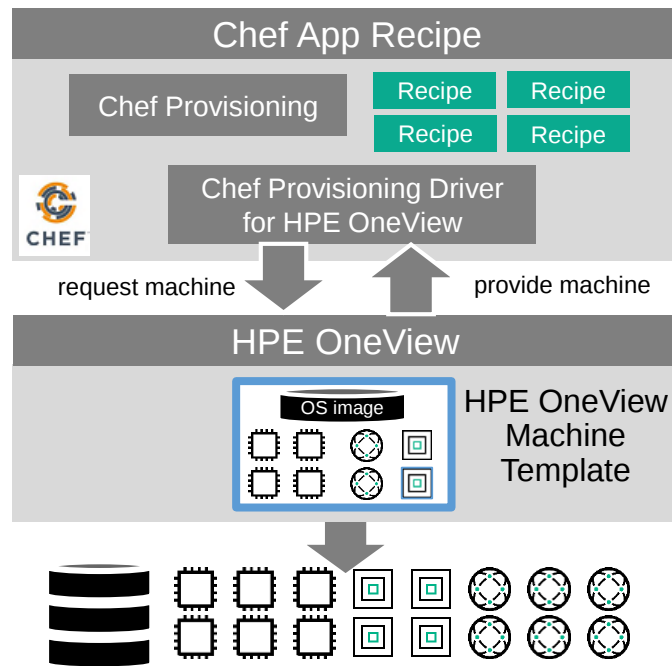
Aprovisionamiento automático de todo el *stack* desde el “puro hierro” hasta las aplicaciones en solo minutos

Incrementar la fiabilidad

Reducir la posibilidad de errores y agilizar la auditoría a través de un aprovisionamiento continuo y consistente

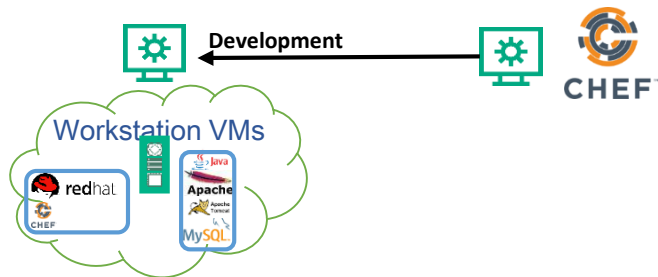
Proporcionar una flexibilidad en el despliegue

Aprovisionar el hardware con una línea de código – de la misma manera que los recursos virtuales y de Cloud



La forma más sencilla de provisionar infraestructura con Chef

Infraestructura como código



```
require 'chef/provisioning/vagrant_driver'

with_driver 'vagrant'

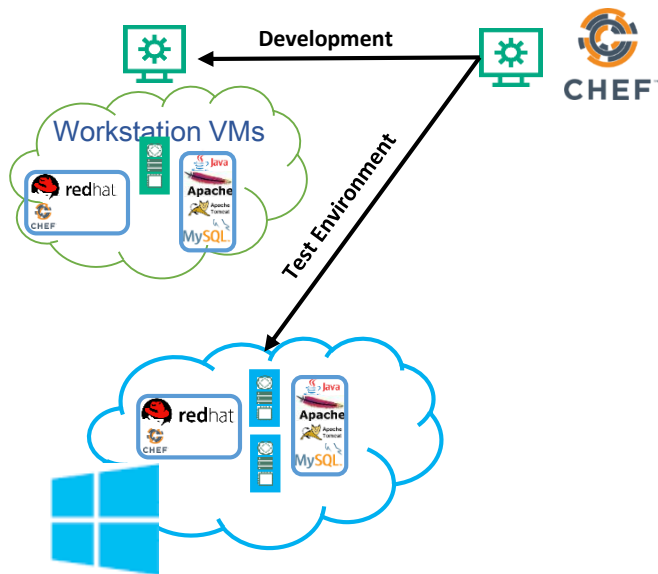
chef_environment 'development'

machine 'web01' do
  machine_options :driver_options => {
    'vm.box' => 'CHEF-RHEL',
    # becomes vm.network(:forwarded_port, guest: 80, host: 8080) in
    # Vagrantfile:
    'vm.network' => ':forwarded_port, guest: 80, host: 8080'
  }

  recipe 'my_apache_webserver'
  converge true
end
```

Vagrant es una herramienta para la creación y configuración de entornos de desarrollo virtualizados. Originalmente se desarrolló para VirtualBox y sistemas de configuración tales como Ansible, Chef, Salt y Puppet.

Infraestructura como código



```
require 'chef/provisioning/azure_driver'

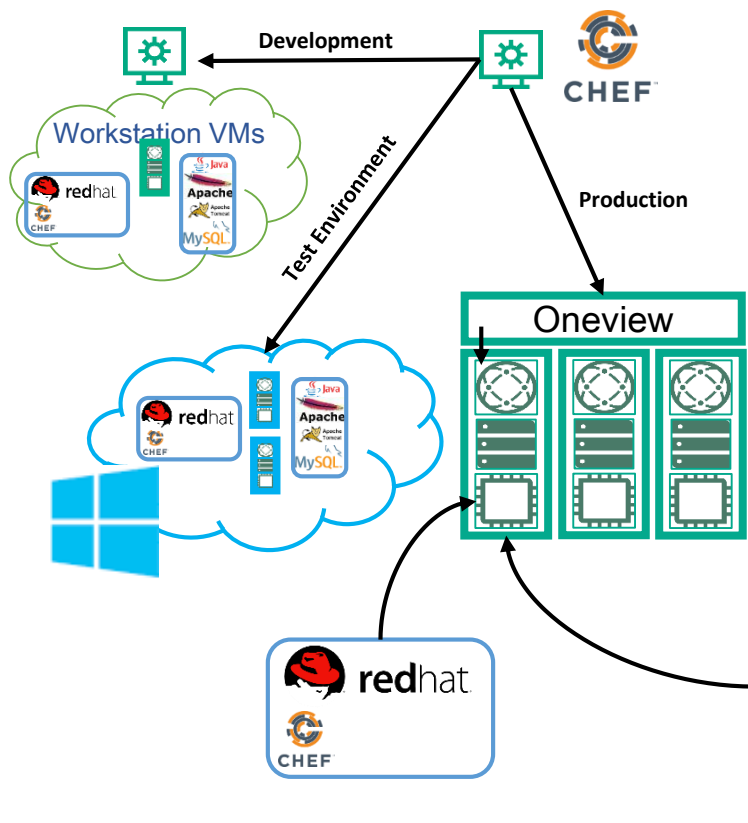
with_driver 'azure'

chef_environment 'test'

machine 'web01' do
  machine_options :driver_options => {
    :bootstrap_options => {
      :cloud_service_name => 'chefprovisioning', #required
      :storage_account_name => 'chefprovisioning', #required
      :vm_size => "Standard_D1", #required
      :location => 'West US', #required
      :tcp_endpoints => '80:80' #optional
    },
    :image_id => 'RHEL-server-20150706-en-us-30GB', #required
    # Until SSH keys are supported (soon)
    :password => "chefm3t41\\m/" #required
  }

  recipe 'my_apache_webserver'
  converge true
end
```

Infraestructura como código



```
require 'chef/provisioning'

with_driver 'oneview'

chef_environment 'production'

machine 'web01' do
  machine_options :driver_options => {
    :server_template => 'Template - Web Server',
    :os_build         => 'CHEF-RHEL-x64',
    :host_name        => 'chef-web01',
    :connections      => {1 => { :dhcp=>true, :net=>'deadnet5', :deployNet =>'pxe5'},
                        2 => { :dhcp=>false, :ip=>'10.0.200.6', :net =>'dmz-prod'}, },
    :domain_type      => 'workgroup',
    :domain_name       => 'synergy.hpe.com',
    :net_mask          => '255.255.254.0',
    :gateway           => '10.0.200.1',
    :dns               => '10.0.200.12, 10.0.200.19, 10.0.200.20' },
  recipe 'my_apache_webserver'
  converge true
end
```

Ansible demo



Ansible playbooks can automate a software stack rolling update

```
- hosts: webserver
  remote_user: root
  serial: 1

  # These are the tasks to run before applying updates:
  pre_tasks:
    - name: disable the server in haproxy
      haproxy: 'state=disabled backend=myapplb host={{
inventory_hostname }} socket=/var/lib/haproxy/stats'
      delegate_to: "{{ item }}"
      with_items: groups.lbservers

  roles:
    - base-apache
    - web

  # These tasks run after the roles:
  post_tasks:
    - name: wait for webserver to come up
      wait_for: 'host={{ ansible_ssh_host }} port=80 state=started
timeout=80'

    - name: enable the server in haproxy
      haproxy: 'state=enabled backend=myapplb host={{ inventory_hostname
}} socket=/var/lib/haproxy/stats'
      delegate_to: "{{ item }}"
      with_items: groups.lbservers
```

Sample playbook.
Performs a non-disruptive
software stack rolling update.

Infraestructura Componible en acción con Docker

La única manera de provisionar infraestructura física directamente desde Docker



Consumidores (Dev)

¿Cómo
construir
más rápido?



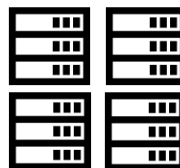
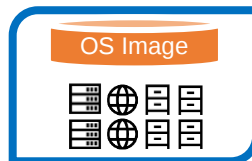
Provisionar hosts desde docker-machine



HPE OneView API

Proveedores (Ops)

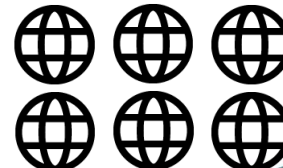
¿Cómo consigo
que funcione?



Servidor



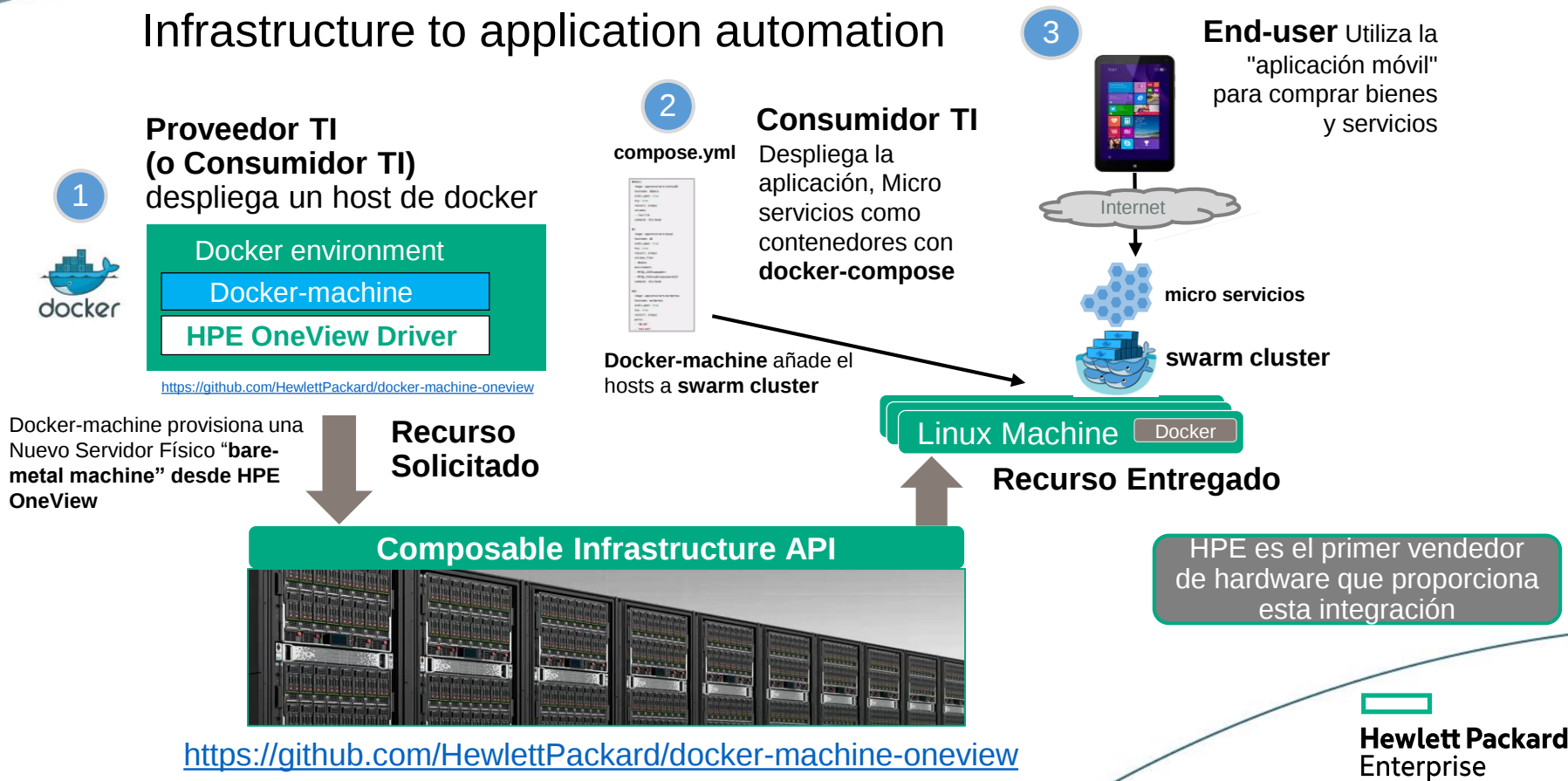
Almacenamiento



Red

Docker-machine driver para HPE OneView

Infrastructure to application automation



Recursos para el Ecosistema de “Composable Infrastructure”

- **HPE OneView Software Development Kit**

- Java SDK: <https://github.com/HewlettPackard/oneview-sdk-java>
- GoLang bindings: <https://github.com/HewlettPackard/oneview-golang>

- **HPE OneView integrations**

- Ansible : <https://github.com/HewlettPackard/ansible-oneview>
- Chef: <https://github.com/HewlettPackard/chef-provisioning-oneview>
- Docker: <https://github.com/HewlettPackard/docker-machine-oneview>
- OpenStack: <https://github.com/HewlettPackard/python-hpOneView>
- Powershell: <https://github.com/HewlettPackard/POSH-HPOneView>
- Python: <https://github.com/HewlettPackard/python-hpOneView>

- **Integrating with HPE OneView: A technical guide for ISVs and Developers**

- <http://www8.hp.com/h20195/v2/GetDocument.aspx?docname=4AA5-8669ENW>

- **HPE OneView REST API Reference**

- <http://h17007.www1.hp.com/docs/enterprise/servers/oneview2.0/cic-api/en/api-docs/current/index.html>

- **HPE OneView documentation**

- www.hp.com/info/oneview/docs



HPE Synergy: La primera plataforma de la industria con arquitectura de componibilidad

Marco “Componible”

Todo lo necesario para ejecutar aplicaciones, la TI se puede instalar y consumir rápidamente

La auto integración hace que el crecimiento sea simple y automatizado a nivel de rack/fila

Listo para Fotónica y Memristor para protección de la inversión

HPE Composer

Inteligencia definida por software integrada para auto-descubrir, auto-integrar y crecer desde racks a filas



Servidores componibles

Proporciona rendimiento, escalabilidad, optimización de la densidad, sencillez de almacenamiento y flexibilidad de configuración



Redes componibles

La conectividad multi-red a nivel de rack elimina switches ToR



Almacenamiento componible

El almacenamiento integrado de alta densidad se puede componer con cualquier servidor (SDS, DAS, SAN)



**Hewlett Packard
Enterprise**

HPE Synergy



Compute Storage Fabric

HPEOVServerProfile -name mysite01 -template ObjectS

Necesito

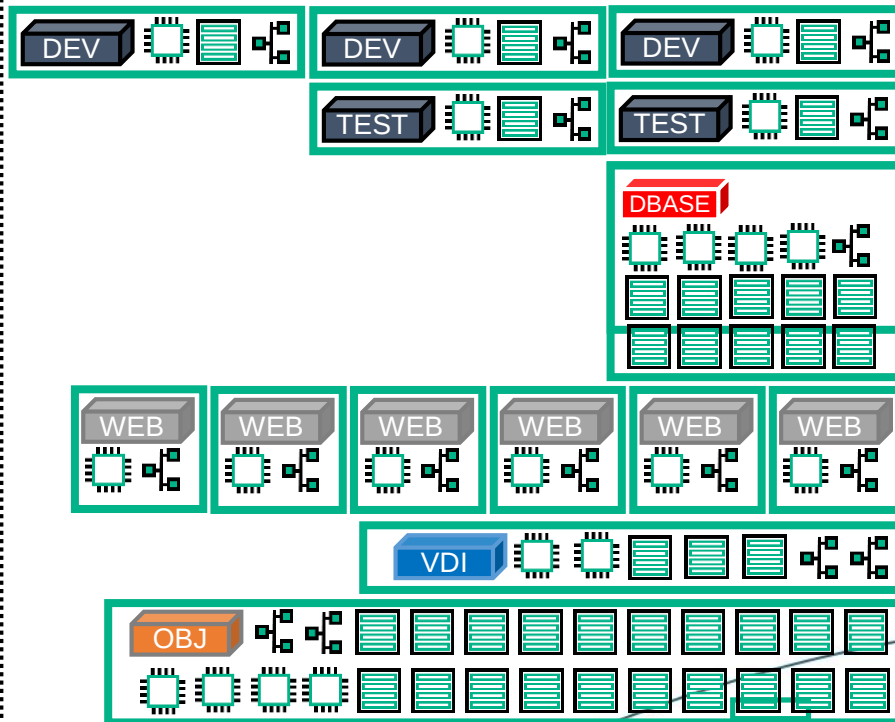
“Entornos de Dev/Test”

“Hacer la **Base de Datos** más grande”

“Web para la temporada navideña”

“**VDI** ahora y en vacaciones termina”

“y **Object Storage**”



Hewlett Packard
Enterprise



Hewlett Packard
Enterprise





**Hewlett Packard
Enterprise**

Muchas Gracias

Pablo.raez@hpe.com

@pabloraez

www.linkedin.com/pabloraez