



Hewlett Packard
Enterprise

HPE Reference Configuration for accelerating DevOps with HPE Synergy and Chef Automate



Reference Architecture

Contents

Executive summary.....	3
Introduction.....	3
Solution overview.....	4
Solution Components.....	5
HPE Synergy.....	5
Software-defined, composable infrastructure	5
HPE Synergy Composer.....	6
HPE Synergy Image Streamer.....	6
Chef Automate.....	7
Provisioning with HPE OneView.....	8
Accelerating DevOps using HPE Synergy and Chef Automate	9
How to deploy a web application and test the entire hardware and software stack for compliance.....	11
Deploying the Chef client via HPE Synergy Image Streamer	11
Server profile templates	12
Visualizing your infrastructure using the Chef Automate dashboard.....	13
Best practices	14
Summary.....	15
Resources and additional links	16

Executive summary

In the era of digital transformation, businesses increasingly rely on new apps and data spanning the data center, cloud, and the edge. As IT becomes an integral part of the business, more and more demands are put on IT to deliver new products, services, and experiences, which is driving unprecedented complexity in the data center. The need for a software-defined infrastructure is emerging where resources can be composed and recomposed from fluid pools of compute, storage, and fabric resources. The Hewlett Packard Enterprise vision of Composable Infrastructure brings together software-defined resources with template-driven provisioning and management to provide the right resource at the right time for any workload and application in the data center. Composability, which brings a public cloud-like experience to your business while improving the experience for your traditional workloads and applications, all using the hardware inside your own data center, is delivered through platforms like HPE Synergy and its built-in management software, HPE OneView.

To make this vision a reality, Hewlett Packard Enterprise has teamed up with several industry-leading configuration management providers, including Chef™ Software. The Chef Automate platform gives developers fast, scalable, and flexible automation of application configuration, deployment, and orchestration. The integration of Chef with HPE Synergy extends the capabilities central to making IT simple. HPE Synergy Composer – powered by HPE OneView – in concert with Chef, gives DevOps teams the ability to automate the provisioning of bare-metal resources, including compute, storage, and networking, as part of the application deployment process. This accelerates time to value through automated, consistent provisioning.

Using HPE Synergy, you can automate IT operations and streamline management of resources, workloads, and apps by deploying and controlling data center infrastructure as code. HPE Synergy Composer, powered by HPE OneView, in combination with Chef Automate reduces risks and increases efficiency by programmatically controlling the management of HPE Synergy infrastructure. The Chef cookbook for HPE OneView managed systems give programmatic access to all aspects of the data center infrastructure, enabling complex application stacks and virtual machines to be quickly built (aggregated, disaggregated, and re-aggregated) and validated. The state and compliance of these elements contained within this infrastructure platform is under continuous monitoring via the Chef Automate platform.

This white paper shows IT architects and administrators how to use software-defined templates through the HPE Synergy Composer to automate provisioning of physical infrastructure and verify their compliance state in conjunction with Chef Automate. HPE continuously collaborates with Chef to gain an understanding on how this current integration can be enhanced to operate seamlessly with Chef's Enterprise Class family of Infrastructure, Compliance and Application Automation tools.

Target audience: This document is intended for IT architects, systems integrators, and partners who are planning to deploy and manage Virtualized servers or containers on a large scale running on Hewlett Packard Enterprise Synergy Composable Infrastructure

Document purpose: The purpose of this document is to describe a best practice scenario for deploying bare metal and virtualized environment on HPE Synergy and automating the full stack with Chef Automate. Readers can use this document to achieve the following goals:

- Gain insight into the value proposition for running Chef Automate in a hybrid virtual and bare metal environment leveraging the strengths of the HPE Composable Infrastructure.
- Learn how to rapidly deploy Systems using Chef Automate and HPE Synergy Image Streamer technology.

Introduction

Succeeding in today's IT world requires IT to evolve from a provider of basic business support into an engine of value creation. Composable Infrastructure in a hybrid environment that empowers IT to create and deliver new value instantly and continuously, accelerating both digital and business transformation. Today, lines of business are putting tremendous pressure on IT leadership to enable new applications and services such as mobile apps, social media sites, cloud services and the Internet of Things (IoT). These applications and services are location-aware and drive real-time, personalized customer experiences and revenue opportunities. At the same time, IT must still manage the traditional applications that run the business – enterprise resource planning (ERP), messaging, databases and more.

The challenge is that these demands require two distinctly different approaches. In traditional IT, change is infrequent; infrastructure is static and applications spin a few times a year at most. With new mobile and cloud-native applications, change delivery is continuous, with many companies spinning application updates daily or even more frequently. To accelerate service delivery, DevOps teams need dynamic infrastructure that can be deployed on demand.

While the public cloud can provide an on-demand option for some new applications, it can also present IT with new enterprise architecture concerns over issues such as security and data compliance. CIOs who want to host applications internally face the need for two separate

infrastructures, one designed for reliability and stability to support mature applications and a second designed for agility and speed to support mobile and cloud-native applications. This is costly and complex, and is inconsistent with another directive that most CIOs confront – to simplify operations and reduce the overall cost of IT.

To address these competing needs, Composable Infrastructure delivers on both models. The concept is simple – physical and virtual resources can be provisioned to any application on-demand regardless of the style of application, or the type, or quantity of the resource.

Think of Composable Infrastructure as a set of flexible building blocks that can be dynamically and automatically composed and recomposed to meet changing workload needs. Composable Infrastructure allows you to deploy IT resources quickly and for any workload using software-defined infrastructure. It can support traditional and cloud-native apps in virtual machines, containers, or on bare metal. It combines software-defined intelligence with a unified API that incorporates powerful tools from the HPE partner ecosystem. It also fully integrates with industry-standard tools such as Chef Automate, Ansible, Salt Stack, Docker, VMware®, OpenStack and others, allowing IT Ops, Operators and Developers to continue leveraging the tools they use every day to speed application and services delivery.

To learn more, read the white paper [HPE Composable Infrastructure: Bridging traditional IT with the Idea Economy](#).

Solution overview

This Reference Configuration provides a solution architected for Chef Automate running on hybrid bare metal and virtualized HPE Composable Infrastructure and integrations to provide automated server provisioning as shown in Figure 1. This creates a scalable and highly available platform for Containers or Virtualized workload deployments.

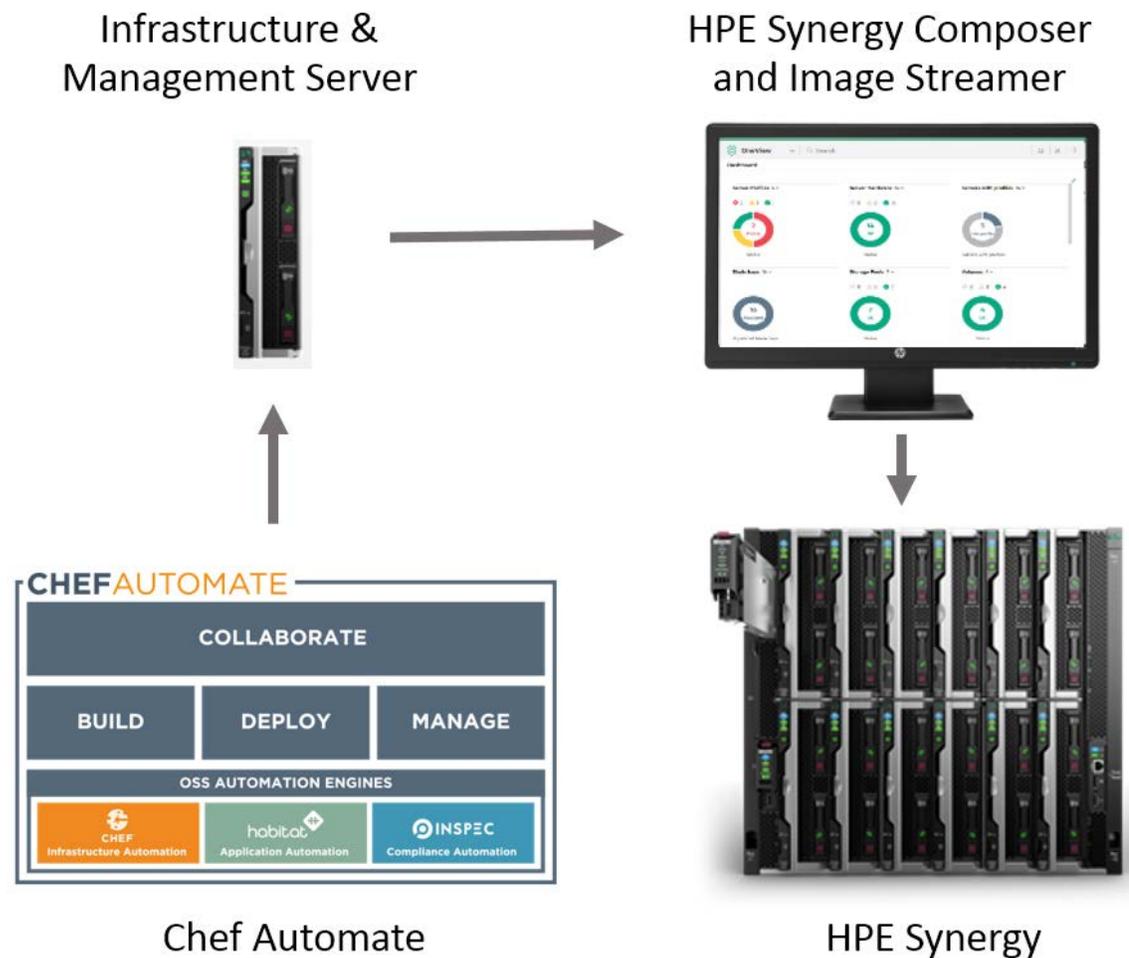


Figure 1. The deployment architecture showing the communication between the Chef Automate server, the infrastructure node and the Synergy system via the Synergy Composer

Solution Components

HPE Synergy

HPE Synergy, the first platform built from the ground up for composable infrastructure, empowers IT to create and deliver new value instantly and continuously. This single infrastructure reduces operational complexity for traditional workloads and increases operational velocity for the new breed of applications and services. Through a single interface, HPE Synergy composes compute, storage and fabric pools into any configuration for any application. It also enables a broad range of applications from bare metal to, virtual machines or containers, and operational models like hybrid cloud and DevOps. HPE Synergy enables IT to rapidly react to new business demands.

HPE Synergy Frames contain a management appliance called the HPE Synergy Composer which is powered by HPE OneView. Together, HPE Synergy Composer and HPE OneView manage the composable infrastructure and deliver:

- Fluid pools of resources, where a single infrastructure of the compute, storage and fabric boots up ready for workloads and demonstrates self-assimilating capacity that can be utilized on demand.
- Software-defined intelligence, with a single interface that precisely composes logical infrastructures at near-instant speeds, and demonstrates template-driven, frictionless operations.
- Unified API access, which enables simple line-of-code programming of every infrastructure element; easily automates IT operational process, and effortlessly automates applications through infrastructure deployment.

With HPE Synergy customers get core benefits including:

- One infrastructure for any application: Eliminate costly silos, free up resources, reduce cost, and simplify IT operations
- Deploy at cloud-like speed and scale: Accelerate application and service delivery
- Automate everyday operations: Increase productivity and control across the data center
- Develop apps faster and smarter: Accelerate your business with a developer-friendly infrastructure

Software-defined, composable infrastructure

Composable infrastructure is a new approach to architecture that allows you to deploy IT resources quickly and for any workload using software-defined infrastructure. It combines this software-defined intelligence with a unified API that incorporates powerful tools from the HPE partner ecosystem, as seen in Figure 2.

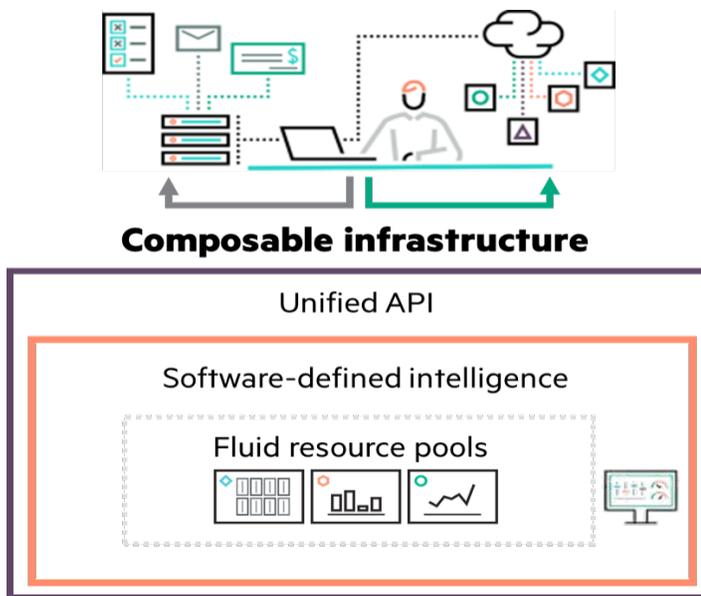


Figure 2. Composable Infrastructure principles

HPE Synergy Composer

The HPE Synergy Frame's unique design embeds HPE OneView in the HPE Synergy Composer to compose compute, storage, and fabric resources in any configuration. HPE Synergy Composer provides the enterprise-level management to compose and deploy system resources based on your application needs. This management appliance uses software-defined intelligence to aggregate compute, storage, and fabric resources in a manner that scales to your application needs, instead of being restricted to the fixed ratios of traditional resource offerings. HPE Synergy can automatically take advantage of HPE OneView's template-based provisioning, enabling fast time to service with a single point for defining compute module state, pooled storage, network connectivity, and boot image.

HPE OneView is an integrated management solution that transforms compute, storage, and networking to software-defined. A single infrastructure management solution increases the productivity of every member of the internal IT team across compute, storage, and networking. By streamlining processes, incorporating best practices, and creating a new holistic way to work, HPE OneView provides organizations with a more efficient way to work. It is designed for open integration with existing tools and processes to extend these efficiencies.

HPE OneView is instrumental for the deployment and management of HPE servers and enclosure networking. It collapses infrastructure management tools into a single resource-oriented architecture that provides direct access to all logical and physical resources of the solution. Logical resources include server profiles and server profile templates, enclosures and enclosure groups (including HPE Synergy Frames), and logical interconnects and logical interconnect groups. Physical resources include computing resources, networking interconnects, and storage resources.

HPE OneView offers a uniform way for administrators to interact with resources by providing a unified API. A unified API enables administrators to easily utilize a growing ecosystem of integrations programmatically, further expanding the advantages of a single management solution to include a larger tool set. It encapsulates and abstracts many underlying tools behind the integrated resource model, so the administrator can operate with new levels of simplicity, speed, and agility to provision, monitor, and maintain the solution.

HPE Synergy Composer provides an interface to manage all HPE OneView resources and is used in all the examples in this whitepaper.

HPE Synergy Image Streamer

HPE Synergy Image Streamer deploys and updates apps and infrastructure in minutes. This management appliance works with HPE Synergy Composer for fast software-defined control over physical compute modules with operating system and application provisioning. HPE Synergy Image Streamer enables true stateless computing combined with the capability for image lifecycle management. This management appliance rapidly deploys and updates OS images to your infrastructure.

HPE Synergy Image Streamer adds a powerful dimension to 'infrastructure as code'—the ability to manage physical servers like virtual machines. In traditional environments, deploying an OS and applications or hypervisor is time consuming because it requires building or copying the software image onto individual servers, possibly requiring multiple reboot cycles. In HPE Synergy, the tight integration of HPE Synergy Image Streamer with the HPE management layer enhances server profiles with images and personalities for true stateless operation.

HPE Synergy Composer, powered by HPE OneView, captures the physical state of the server in the server profile and enables a software defined approach to physical hardware using server profiles. HPE Synergy Image Streamer enhances this server profile (and its desired configuration) by capturing your golden image as the 'deployed software state' in the form of bootable image volumes. These enhanced server profiles and bootable OS plus application images are software structures ('infrastructure as code')—no compute module hardware is required for these operations. The bootable images are stored on redundant HPE Synergy Image Streamer appliances, and they are available for fast implementation onto multiple compute nodes at any time. This enables bare metal compute modules to boot directly into a running OS with applications and multiple compute nodes to be quickly updated.

Chef Automate

Businesses that embrace apps and digital business as the new customer interface will be the leaders in the coming years and will outperform their peers, industries, and markets.

To become a disruptor and not be one of the disrupted, businesses must deliver software at speed, with efficiency and low risk. Amid this need for speed, then there are other disruptive changes:

- The shift to cloud-native, modern applications that are designed for scale while still managing a portfolio of legacy applications.
- The shift toward container-based runtimes for applications composed of multiple micro services while still managing monolithic stateful applications.
- The shift to vertically integrated development teams. Starting with Agile, Lean and DevOps (ALDO) techniques some silos are being broken by modern application teams already. But more is needed to better integrate all stakeholders: from the business to compliance.

Delivering software at speed during these shifts requires a platform that can deliver optimized ALDO processes, and consistent infrastructure, application, and compliance automation regardless of the IT environment. Chef Automate is the leading platform for this 'Continuous Automation' approach. Chef Automate can support any environment, from applications that run on bare metal in the data center to container based micro services in the cloud.

Chef Automate is powered by three open source engines: Chef, Habitat and InSpec, as shown in Figure 3:

- **Chef** is the engine for infrastructure automation.
- **InSpec** lets you specify compliance and security requirements as executable code.
- **Habitat** automates modern applications such as those that are in containers and composed of micro services.

To automate their infrastructure, Chef Administrators write declarative scripts, called recipes, which are stored in cookbooks. These cookbooks can be used to automate software processes. Chef recipes are more efficient and reliable than shell scripts or manual processes due to them being Repeatable, Testable and Versionable. Chef recipes are easy to write and share among IT organizations through the Chef Supermarket.



Figure 3. The components of Chef Automate

Provisioning with HPE OneView

To create an agile and efficient IT infrastructure, organizations must transform traditional rigid physical systems into flexible, physical or virtual, resource pools. As they do this, they are often challenged to automate the provisioning of the physical “bare metal” hardware. Provisioning is a complex process, requiring configuration of hardware, connectivity, OS, and application stacks on the right resources with the right configurations. A typical provisioning process might include these steps, as shown in Figure 4:

1. Update firmware
2. Update drivers
3. Set BIOS settings
4. Set unique identifiers (WWN, SN, UUID, MAC)
5. Install OS
6. Configure smart array
7. Configure network connectivity
8. Configure SAN zoning
9. Configure SAN array

Traditional process—needs a different tool for every task

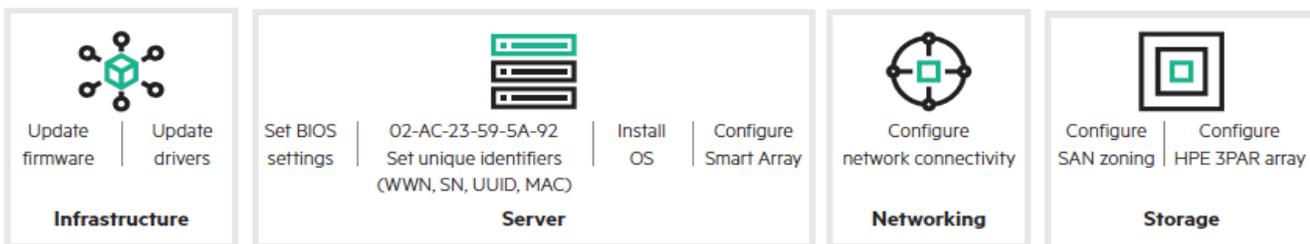


Figure 4. Traditional multi-step process for provisioning hardware is inefficient

Using traditional processes, standing up infrastructure can require three or four independent teams (compute, storage, networking, and more). These teams use several different tools to execute as many as 50 or more separate scripts consisting of thousands of lines of code; all of which can require manual intervention and monitoring along the way. The process is time consuming and error prone, and it can take days or weeks to integrate and stand up a new server.

Composable Infrastructure, delivered through HPE Synergy, allows IT to create re-useable profile templates to define infrastructure configuration and settings across compute, storage, and networking, which can then be provisioned from bare-metal resources through a single line of code. Templates created in HPE OneView allow IT to create logical infrastructures from available resource pools, specific to each application’s needs.

HPE OneView utilizes a unified API to provide a consistent interface for discovering, searching, inventorying, configuring, provisioning, updating, and diagnosing composable resources. Using rich, high-level infrastructure constructs, it eliminates the need for multiple, time-consuming, low-level programming interfaces so administrators and software developers can be more productive.

In addition to initial provisioning, change operations such as adding additional storage to a service, modifying network connectivity, or updating firmware can also be implemented via templates. The use of repeatable templates allows changes to be deployed automatically and significantly reduces manual interaction and errors.

Instead of using dozens of scripts to automate physical infrastructure, Composable Infrastructure lets IT admins and DevOps teams use just a single line of code to orchestrate all the steps needed to set up a physical infrastructure. The unified API makes it possible to treat physical infrastructure as code.

Accelerating DevOps using HPE Synergy and Chef Automate

Chef Automate enables rapid and reliable deployment and updating of HPE Synergy, using recipes that can be versioned and tested just like application software, as shown in Figure 5. HPE Synergy Composer brings infrastructure as code to bare metal through templates that unify the process for provisioning compute, connectivity, storage and OS deployment into a single step, just like provisioning VM instances from the public cloud. Together, they provide a software-defined approach to the management of the entire data center giving IT the ability to deliver new or updated services on an as needed or on-demand basis.

The Chef Cookbook for HPE OneView provides an interface to manage all HPE Synergy resources and is used in all the examples in this whitepaper. With it, an administrator can describe all the resources which are required for the workload. This means that in the data center, compute, storage and fabric resources can be provisioned quickly, transparently and reliably. Speed and efficiency can be gained without compromising the security of sensitive data, which stays controlled within the physical network.

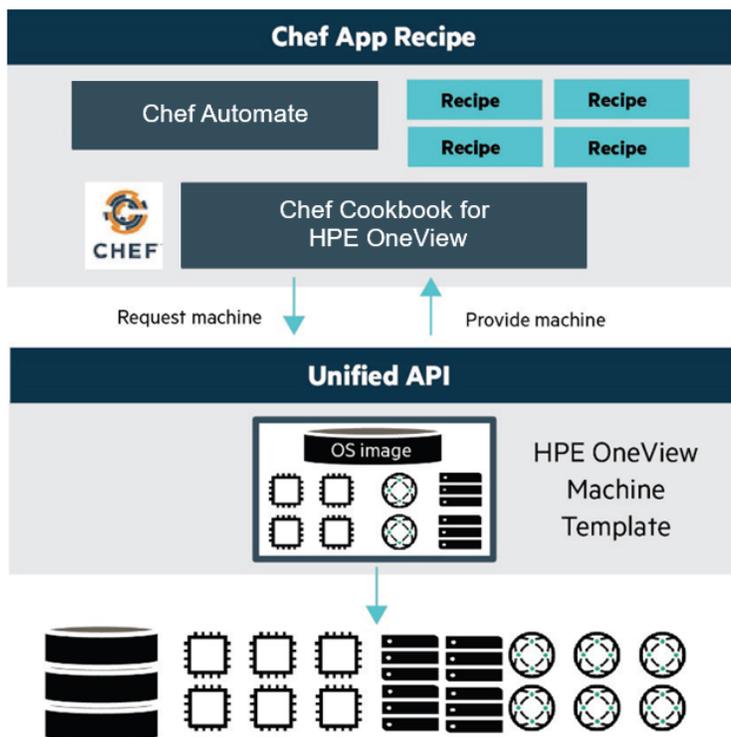


Figure 5. Overview: using HPE OneView to provision infrastructure from Chef Recipes

To take advantage of the integration between Chef Automate and HPE Synergy, an administrator develops a recipe with the following steps.

1. Provision a physical compute module using HPE OneView template with HPE Synergy Composer
2. Deploy operating system using an OS build template from HPE Synergy Image Streamer, the HPE OneView OS deployment appliance Image Streamer
3. Configure the application stack

Example Deployment Scenario:

Here's a scenario which may be familiar to many Operators. Running most modern applications involves three steps:

1. Set up a network to serve up the application's traffic
2. Provision compute resources to run the application
3. Build storage to retain data and content

In traditional IT environments these steps are commonly handled by completely separate IT teams, which requires a complex process to coordinate.

1. Open a support ticket to request a new environment
2. Review and approval of the request by the change management committee at the following weekly meeting
3. Forward the ticket to three IT teams: Network, Storage, and Systems who schedule the task according to their workloads
4. Create and make available new disk LUNs: Storage team
5. Provision of ports and IP addresses: Network team
6. Installation of operating system and ensuring the network and storage are configured correctly: Systems team
7. Review, accept and sign-off of the support ticket
8. If steps #4, #5, or #6 are incorrect or incomplete, further support ticket requests will be needed until all issues are resolved

It may take up to two weeks for the application developer to be able to log on to the environment and begin their work. Depending on the complexity of the organization, the delivery of virtual machines can take two months or more.

The silo structure inevitably creates bottlenecks leading to avoidable delays and frustrated users. How can a business respond? Creating “shadow IT” environments not managed by IT teams using company credit card and public cloud provider environments is insecure, unregulated and expensive. Moving everything into the cloud is usually not a viable option for legacy applications and workloads. Modern infrastructure management gives an efficient solution.

HPE and Chef have partnered to bring ‘infrastructure as code’ down to the layer of bare metal. Here’s an example of network configuration. To create a new Ethernet network for an application in a traditional data center would usually require logging onto a network device and running commands to create the network and to add all the correct routing data. A simple Chef recipe resource like below will handle this:

```
oneview_ethernet_network 'Eth1' do
  client my_client
  data(
    vlanId: 1001 purpose: 'General'
    smartLink: false
    privateNetwork: false
  )
end
```

A new storage pool could be added by the following:

```
oneview_storage_pool 'CPG_FC-AO' do
  client my_client
  storage_system 'ThreePAR7200-8147'
end
```

For compute, the following example would bring up a new physical server in enclosure 1, bay 2 of the Synergy Frame:

```
oneview_server_profile 'ServerProfile2' do
  client my_client
  data(
    description: 'Override Description',
    boot: {
      order: [],
      manageBoot: true
    }
  )
  server_profile_template 'ServerProfileTemplate1'
  server_hardware 'Encl1, bay 2'
end
```

The concept of Chef Recipes is “executable documentation”; start by designing your application stack, then write the simple building blocks, or resources, for each of these parts, and string these together. There are many more examples of Chef Resources for OneView available on GitHub at <https://github.com/HewlettPackard/oneview-chef-automate>.

How to deploy a web application and test the entire hardware and software stack for compliance

This walk-through demonstrates how to provision a new application from bare metal including deployment of the base operating system and deployment of an application on the server. In this case we show the deployment for a simple web application running in a bare metal server on an HPE Synergy Frame.

The walk-through assumes that:

- HPE Synergy Composer is present with browser access to the HPE Synergy Composer UI.
- HPE Synergy Image Streamer is available and has an OS Deployment Plan for Red Hat Enterprise Linux 7.3 available.
- Server profile templates have been created in HPE Synergy Composer to provision servers.
- There is an available Synergy Frame and Synergy module.
- A Chef Automate server is present with browser access to the Chef Automate UI.
- A node is attached to Chef Automate which is used to configure the HPE Synergy Frame via the HPE OneView API.

Deploying the Chef client via HPE Synergy Image Streamer

We are using the HPE Synergy Image Streamer to deploy the operating system image to the server. In this example that is a standard Red Hat Enterprise Linux 7.3 Operating System image. This only installs and configures the base operating system; what we installed needs one more step before the workload is ready to run. We modify the OS Build plan within HPE Synergy Image Streamer to additionally install and run the Chef client as part of the OS deployment. This is done by adding a bash script to the OS Build plan that bootstraps the machine with Chef and set it up correctly. This step, as shown in Figure 6, allows the Chef client to install the relevant packages and middleware once the OS is set up.

Used by [2 deployment plans](#)

General

Steps [Edit](#)

Order	Name	Type
1	RHEL-7.3-mount-and-validate-2017-03-15	General
2	RHEL-7.3-configure-users-2017-03-15	Deploy
3	RHEL-7.3-configure-multiple-NIC-teaming-2017-06-02	Deploy
4	RHEL-7.3-configure-hostname-2017-08-21	Deploy
5	RHEL-7.3-manage-security-services-2017-04-20	Deploy
6	RHEL-7.3-configure-partition-using-LVM-2017-03-15	Deploy
7	Install Chef	Deploy
8	RHEL-7.3-unmount-2017-03-15	General

Figure 6. An OS build plan in HPE Synergy Image Streamer showing the step to install and configure Chef

Server profile templates

The server profile template describes how the server will run in the data center, including firmware versions and settings, local and SAN storage configurations, boot settings, BIOS settings, network connections and more. The server profile templates allow you to capture the best practices for different types of workloads in a repeatable format.

Server profile templates are managed in HPE Synergy Composer. First, we select a server template to include in your Chef recipe. In this case we're selecting a template that uses an OS Deployment plan from the HPE Synergy Image Streamer, in this case the 'Red Hat 7.3' template.

Here's an extract from the recipe showing the definition of the server via a server profile and how we power on the server:

```
oneview_server_profile 'Chef-Node-1' do
  client my_client
  server_hardware 'BOT-CN75150107, bay 4'
  server_hardware_type 'SY 480 Gen9 CNA Only'
  enclosure_group 'EnclosureGroup1'
  server_profile_template 'RedHat 7.3'
end

oneview_server_hardware 'BOT-CN75150107, bay 4' do
  client my_client
  power_state 'on'
  action [:set_power_state, :update_ilo_firmware]
end
```

Note

The fine-grained control this Chef recipe gives over the hardware and software. The new compute module is assigned in the bottom rack of the enclosure in bay 4 and the firmware is updated, as you can see in Figure 7. You can see the entire recipe, where we also create the related networking and storage resources, on GitHub at https://github.com/HewlettPackard/oneview-chef-automate/blob/master/HPE_RC_Chef/hpe_oneview_synergy.rb.

HPE OneView

The screenshot displays the HPE OneView interface for a server named 'BOT-CN75150107, bay 4'. The server is in the 'Powering On' state. The hardware details include Synergy 480 Gen9, SY 480 Gen9 CNA Only type, and serial number CN7518002Y. Utilization shows 0% CPU, 0 W power, and 70 °F temperature.

Hardware >		Utilization >	
Server name	tesx1.mgmt.local	CPU	0% of 24 cores
State	Profile Applied	Power	0 W of 206 W
Server profile	● Chef-Node-1	Temperature	70 °F
Server power	Powering On		
Model	Synergy 480 Gen9		
Server hardware type	SY 480 Gen9 CNA Only		
Serial number	CN7518002Y		
iLO address	ILOCN7518002Y		

Figure 7. The Composer UI shows the new server being built on the bottom rack in bay number 4 when the script is run

We have used a simple Chef recipe to provision a physical infrastructure completely including loading firmware, configuring a BIOS, creating networking and storage connections, and installing an OS. This collapses hundreds of lines of code into just a few, letting admins configure physical infrastructure using the code in the same way as they might configure cloud-based infrastructure.

In this example an instance of the company's ecommerce site is stood up which is a Java application server and a MySQL database but the same OS deployment plan can be used for any workload by changing the Chef run list that defines the cookbooks and recipes that are required for the specific workload.

Visualizing your infrastructure using the Chef Automate dashboard

Once the server is running we can also check its status in the Chef Automate dashboard, as seen in Figure 8. The Converge Status dashboard show two nodes – the infrastructure node that configures and manages the HPE Synergy Frame, and the newly built ecommerce server. The diagnostics indicate the Chef recipes are running successfully, which means everything was built exactly to code. From this point forward, the Chef client will run every 30 minutes on each server, which will quickly catch and remediate any configuration drift on the server.

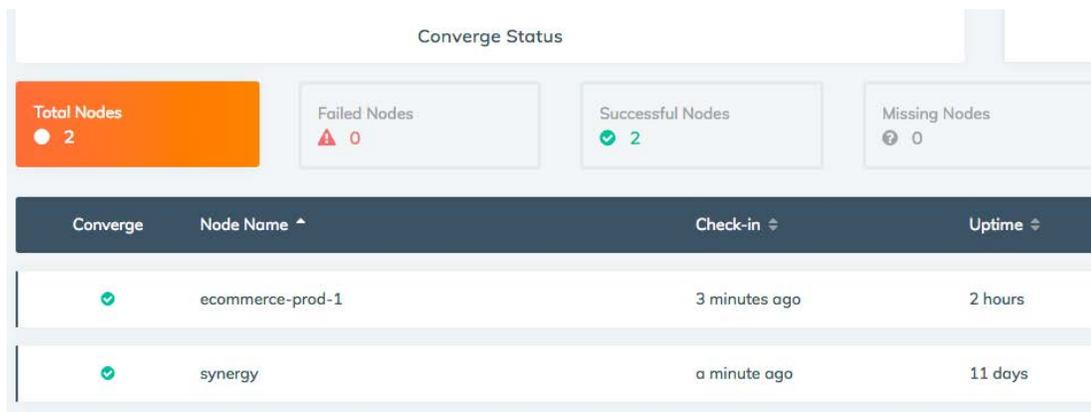


Figure 8. The Chef Automate UI shows the status of the HPE Synergy configuration we applied along with the ecommerce server we created.

Chef Automate also has the ability to scan the entire fleet for compliance against a wide set of compliance standards. This is powered by Chef's compliance-as-code tool, InSpec. InSpec is a language and framework for describing audit, security, compliance and QA requirements, as seen in Figure 9.

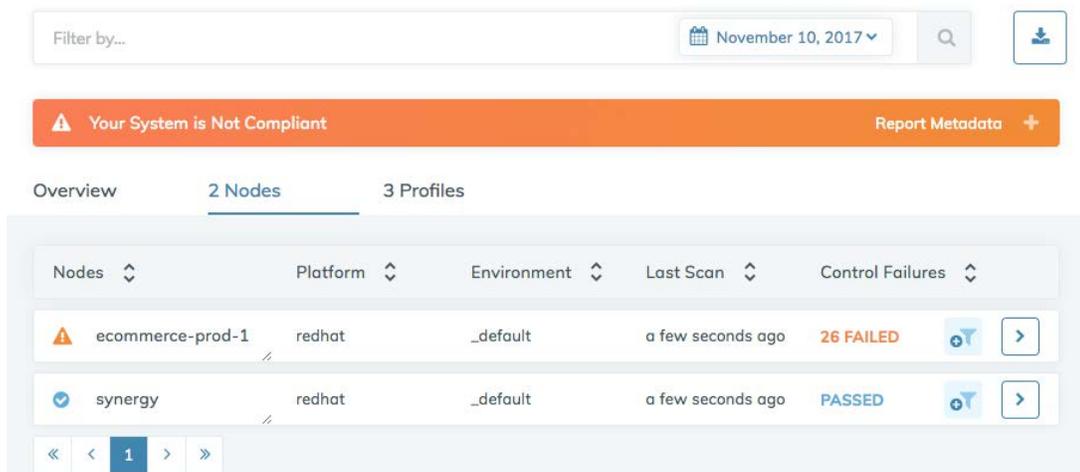
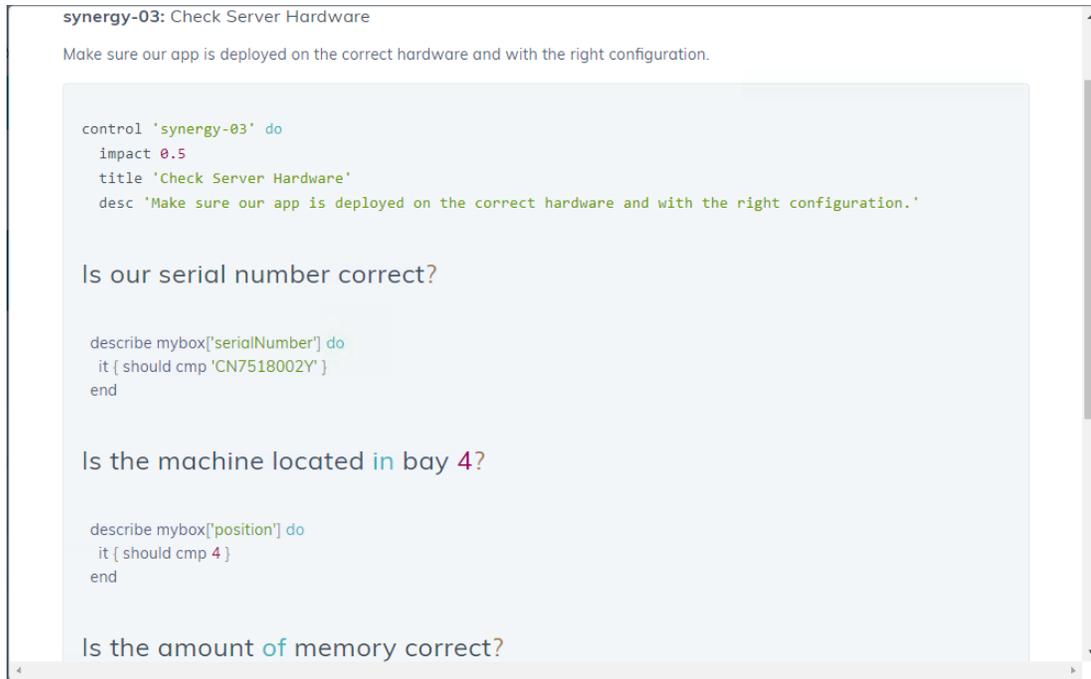


Figure 9. The Chef Automate UI shows the status of the HPE Synergy node where compliance tests are failing.

In Figure 10, InSpec is used to ensure that all infrastructure is built correctly and stays compliant with internal and external security regulations. We have written InSpec code to make sure the application was deployed onto correct hardware. This is deployed and executed by the Chef client.

The image shows a screenshot of a code editor window titled "synergy-03: Check Server Hardware". The window contains InSpec code for checking server hardware. The code is as follows:

```
control 'synergy-03' do
  impact 0.5
  title 'Check Server Hardware'
  desc 'Make sure our app is deployed on the correct hardware and with the right configuration.'

  Is our serial number correct?

  describe mybox['serialNumber'] do
    it { should cmp 'CN7518002Y' }
  end

  Is the machine located in bay 4?

  describe mybox['position'] do
    it { should cmp 4 }
  end

  Is the amount of memory correct?
```

The code is displayed in a light blue background with a white border. The text is in a monospaced font, with some words highlighted in green and blue. The window has a scrollbar on the right side.

Figure 10. InSpec code to check the server was deployed by the Synergy Composer to the right hardware configuration

Best practices

A robust, automated data center takes time to organize, build, maintain, and support. Here are some best practices you should consider for your environment.

Divide your work

When creating Chef Recipes, don't let them get very big; instead, maintain small recipes that serve specific purposes. In doing so, consider the likelihood that the recipes you're creating will need to be used again. If you're doing a large amount of provisioning, it's very likely you will need to reuse them. If you want, you can even share them with others.

Don't hardcode

Rather than hard-coding parameters, code your Chef recipes to be generic and put your parameters into an environment definition, or an attribute file. Then, when running the recipe, you can simply make changes to one of these resources rather than needing to re-code your recipe.

Don't reinvent the wheel

Chef has an enormous number of features. It was built from the ground up specifically for application and infrastructure automation, ranging from individual servers to huge, globally dispersed data centers. No matter what type of work you're trying to accomplish, Chef can probably handle it, either natively or through simple scripts that use the native functionality. And with thousands of users and a global community sharing best practices and source code, it is quite possible that the code you need already exists in the [Chef Supermarket](#) or from other sources such as [GitHub](#).

The Chef Supermarket is filled with thousands of cookbooks built by avid Chef users. It provides useful searching features to narrow results by platform (such as Windows®, Red Hat®, or Ubuntu), and is a great place to see who the authors are, how many times the cookbook has been downloaded, and often provides links to the source code so you can go inspect it for yourself. Note that cookbooks are not compiled, so you should always examine the code before using it and deploying it. If the code is hosted publicly on sites like GitHub, you should also check the reported issues and look at the recent activity to see if the cookbook is being actively maintained and issues are being addressed.

Hewlett Packard Enterprise is proud to support the open source community, and has made our Chef recipes available on GitHub at github.com/HewlettPackard/oneview-chef. We have built a community around our recipes, and encourage other developers to fork them, modify them, add new features, and then perform a pull request so we can add the features back in. Also, all the examples used in this paper are available on GitHub at github.com/HewlettPackard/oneview-chef-automate.

Consider Compliance up front

Often the task of assessing the compliance of your infrastructure is only considered at the last moment of building a production system. By describing your compliance-as-code and testing it as early as possible in the development cycle you reduce uncertainty when it comes time to audit your system. InSpec makes it easy to describe your compliance requirements and run them against your infrastructure.

Summary

In the era of digital transformation IT leaders must find new and innovative ways to deliver unparalleled flexibility and scalability to the business while reducing complexity and operating costs. HPE Composable Infrastructure, delivered through HPE Synergy, enables IT to deploy resources quickly for any workload and application, and manage infrastructure as code.

A key tool in providing infrastructure-as-code is Chef Automate, which provides programmers, development teams, and operations engineers the common toolbox they need to take advantage of the distributed and networked nature of modern applications. HPE Synergy and the HPE Synergy Composer, powered by HPE OneView, together with Chef Automate, gives your DevOps teams true infrastructure-as-a-code capabilities within your data center, supporting faster time to value for new revenue-generating applications that elevate and enhance the business..

Resources and additional links

HPE Reference Architectures, hpe.com/info/ra

HPE Synergy Reference Architectures, <https://hpe.com/info/synergy-ra>

HPE Synergy, hpe.com/synergy

HPE Technology Consulting Services, hpe.com/us/en/services/consulting.html

Cookbook for HPE OneView, supermarket.chef.io/cookbooks/oneview

One Chef Automate GitHub, <https://github.com/HewlettPackard/oneview-chef-automate>

Chef, <https://www.chef.io/>

To help us improve our documents, please provide feedback at hpe.com/contact/feedback.

Learn more at

hpe.com/info/composableinfrastructure



Sign up for updates

© Copyright 2018 Hewlett Packard Enterprise Development LP. The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

The Chef™ Mark and Chef Logo are trademarks/ service marks of Chef, in the United States and other countries and are used with Chef Software Inc.'s permission. Microsoft and Windows are registered trademark of Microsoft Corporation in the United States and other countries. VMware® is a registered trademark of VMware, Inc. in the United States and/or other jurisdictions. Red Hat is a registered trademark of Red Hat, Inc. in the United States and other countries.

