Compliments of:

vmware®

Aria Automation

for dummies®
A Wiley Brand

Embrace an agile infrastructure

Enhance your processes with automation

Enjoy cost savings for your business

Alina Thylander and Nicole Sholly

Francisco Hernandez, Vincent Riccio, Sam McGeown and Karl Fultz

VMware Special Edition

About VMware

VMware offers a breadth of digital solutions that powers apps, services, and experiences which enable organizations to deliver the best customer service and empower employees. VMware streamlines the journey for organizations to become digital businesses that deliver better experiences to their customers and empower employees to do their best work. Our software spans App Modernization, Cloud, Networking & Security, and Digital Workspace.

Since our founding in 1998, our employees and ecosystem of over 30,000 partners have been behind the technology innovations transforming entire industries — from banking, healthcare, and government to retail, telecommunications, manufacturing, and transportation.

Every day, we work to solve our customers' toughest challenges through disruptive technologies like edge computing, artificial intelligence, blockchain, machine learning, Kubernetes, and more — to define the digital foundation that will accelerate the next wave of innovation.

vmware.com

Aria Automation

VMware Special Edition

by Alina Thylander and Nicole Sholly

Contributors: Francisco Hernandez, Vincent Riccio, Sam McGeown, and Karl Fultz

for dummies®
A Wiley Brand

Aria Automation For Dummies®, VMware Special Edition

Published by
John Wiley & Sons, Inc.
111 River St.
Hoboken, NJ 07030-5774
www.wiley.com

ISBN 978-1-394-23466-0 (pbk); ISBN 978-1-394-23467-7 (ebk)

Publisher's Acknowledgments

Development Editor: Rachael Chilvers

Project Editor: Saikarthick Kumarasamy

Acquisitions Editor: Traci Martin

Editorial Manager: Rev Mengle

Business Development Representative: Cynthia Tweed

Introduction

In today's rapidly evolving digital landscape, staying ahead requires organizations to embark on infrastructure modernization initiatives. However, this effort is not without challenges. Managing diverse technologies, overcoming human resistance to change, and ensuring profitability can pose demanding obstacles for any successful IT strategy.

Organizations also need to adjust their business tactics and prioritize funds to keep their competitive advantage, or cash in on new opportunities with new income streams. IT teams, which have predominantly been affected by an increased need for digital resources, must also redesign their tactics to help maintain and drive innovation.

This creates an opening, particularly for IT infrastructure and operations teams that maintain the needs of developers and DevOps engineers, to streamline IT delivery processes by embracing modern automation practices. Because the existing IT infrastructure processes have been too tightly coupled and too inflexible to maintain modern application development practices, developers have embraced public clouds and open-source tools without a lot of IT oversight. Therefore, modernizing IT processes with automation holds the key to empowering IT to drive innovation together with developers in a safe, compliant, agile, and scalable way.

About This Book

Aria Automation For Dummies, VMware Special Edition, helps you understand how you can simplify your cloud experience with Aria Automation.

This handy guide consists of eight chapters to help you navigate network automation in your organization. The chapters cover the following:

- The need for automation and an introduction to Aria Automation (Chapter 1)
- A look at the key components of Aria Automation (Chapters 2–6)
- A glance at the use cases (Chapter 7)
- Ten ways to start with automation (Chapter 8)

Foolish Assumptions

When writing this book, we make the following assumptions about you, the reader:

- You know the basics about automation
- You're familiar with the concept of virtualization
- You understand at a high level the infrastructure-as-code concept
- You're aware of many DevOps processes and tools

Icons Used in This Book

We use icons throughout the book to highlight important information.


TIP

Look to the Tip icon to learn shortcuts and other information that can make your life easier.


REMEMBER

The Remember icon flags facts that are especially important to know.


TECHNICAL STUFF

This icon lets you know that we're about to dole out in-depth techy info. Feel free to skip these if you don't need the details.

Beyond the Book

You don't need to read this guide from cover to cover. Just jump to a specific chapter that interests you and start reading. You can go back later and read any chapters you skipped to make sure you don't miss any vital information.

For the latest news and information, visit `www.vmware.com/products/aria-automation.html`.

IN THIS CHAPTER

- Learning the need for automation
- Introducing the modern, event-driven infrastructure automation platform
- Discovering the Aria Automation Advantage

Chapter 1
Introducing Aria Automation

Automation is not just a buzzword. With the potential to set the foundation for major efficiencies, process enhancements, and cost-cutting strategies, automation tools have become critical for organizations looking to modernize IT and maintain a competitive edge.

VMware has moved toward an application-oriented, developer-driven position, which will be extremely important as the infrastructure and its management become more invisible. Software and application delivery that take advantage of cloud-native constructs will be essential to the success of digital transformations, with the infrastructure portion gradually moving into the cloud.

This book uncovers VMware's modern, event-driven infrastructure automation platform, and it provides you with the opportunity to discover why it's a great solution for every enterprise.

Understanding the Need for Automation

Before we move forward, can we all agree that building and managing a multi-cloud environment to support the needs of IT admins, cloud admins, DevOps teams, lines of business, and developers is complicated? Traditionally, the IT landscape has struggled with widespread human error, over-provisioned resources, and poor governance. The lack of automation led to failed deployments, compliance and security risks, below par performance, and downtime issues. Moreover, IT teams couldn't keep up with business needs as legacy manual processes slowed application delivery and developers found themselves waiting weeks for their environments to function properly.

So, what does it take for IT to pivot away from traditional workload management toward a more agile infrastructure and application configuration, provisioning, DevOps life cycle operations, orchestration of cloud, and software-defined data center environments? Automation is at the heart of this, with a bold new approach to technological practices.

Welcoming the Modern, Event-Driven Infrastructure Automation Platform

REMEMBER

In today's world, business agility is driven by software agility, and all software needs to run on a cloud for an efficient provisioning of the underlying infrastructure. The cloud can be public, private, or hybrid, and it must have various key attributes, such as security, self-optimization, resilience, self-service, and scalability, and it must be rich with services.

We often hear that IT admins spend too much time building and operating the IT infrastructure to avoid downtime, data loss, and security breaches, and to prevent performance problems for consumers. This triggers the need for additional staff to handle mundane, repetitive tasks, and it increases pressure from management, audits, and late-night calls.

Similarly, cloud admins have difficulty managing service offerings without a centralized view across infrastructure resources and applying them across the environment. This leads to

productivity issues due to lack of security and governance, affecting credibility and relevance to DevOps teams. DevOps engineers struggle to keep up with their SLAs. They are on call for downtime in production and need to be able to identify and troubleshoot events that can affect availability and performance of the software. Common challenges they must deal with include processes that are insufficiently automated and not transparent across the infrastructure and app life cycle.

The decision makers struggle to maintain relevance in a changing world and combat the perception of legacy and outdated IT, which leads to reduced DevOps credibility. They're bogged down with multi-cloud deployment models, technology stacks using loosely connected open-source components, and complex automation workflows. As a result, they're unable to balance business agility, business efficiency, and risk mitigation.

Ultimately, end users must find ways to solve these issues to achieve faster time to market, improve customer satisfaction, and increase throughput and business innovation capabilities. Likewise, decision makers need to assess the impact on cloud costs, security risks, and business interruption.

To address these challenges, we introduce VMware Aria Automation, which is a modern event-driven infrastructure automation platform powered by DevOps principles and open-source innovation, delivering a self-service consumption and delivery layer across VMware Cloud and public clouds.

REMEMBER

Aria Automation includes the following components:

- **Assembler:** Infrastructure as Code (IaC) engine for templating, multi-cloud provisioning, and network automation.
- **Service Broker:** Self-service catalog with policy and governance, including support for infrastructure pipelines and Kubernetes infrastructure.
- **Pipelines:** Infrastructure and application deployment automation, along with infrastructure pipelining.
- **Orchestrator:** Workflow automation platform that simplifies and automates complex data center infrastructure processes.
- **Config:** Intelligent automated software configuration management tool that can manage, secure, and optimize any infrastructure — on-premises, in the cloud, or at the edge.

REMEMBER

These components aren't released or sold individually; they're only available as part of Aria Automation.

Identifying the Aria Advantage

To embrace the digital transformation, businesses need technology that adapts and scales rapidly, reliably, and cost-effectively. Applications are more important than ever, progressively serving as the means by which organizations engage with customers. As a result, enterprises require more agility and speed throughout their digital processes, including the workflows involved in the designing, deployment, and management of applications.

Designed to deliver self-service clouds, multi-cloud automation with governance, and DevOps-based infrastructure management and security, Aria Automation helps improve IT agility, productivity, and efficiency, so you can prepare for the future of your business.

REMEMBER

Key benefits delivered by Aria Automation include:

- **Fast time to market:** Automate infrastructure delivery and offload manual tasks with advanced workflows and agile templating.
- **Ease of use:** Rapidly set up and manage multi-cloud environments throughout the life cycle with a native, self-service consumption experience.
- **Security and control:** Establish consistent policies across multi-cloud environments and strengthen infrastructure with native compliance management, flexible guardrails, and vulnerability remediation.
- **Accelerated innovation:** Expedite the adoption of new, cloud-native technologies with a future-ready platform built on IaC and DevOps principles.

Aria Automation provides an automated infrastructure solution to central IT departments and DevOps teams in public sector, education, commercial, and enterprise companies across all industries and geographies with vSphere-based virtualized data center, multi-cloud environments, and extending to public clouds. Some of the most noteworthy capabilities include:

- **Self-service provisioning for multi-clouds:** Unified and consistent IaaS consumption layer with a self-service catalog that enables users to request and provision infrastructure resources across VMware Cloud infrastructure or any major public cloud with IaC.
- **Centralized policy and control:** Flexible guardrails including role-based policies across all cloud environments to maintain proper security and compliance.
- **Cloud agnostic templating:** VMware Aria Automation Templates support for workloads abstracted from the underlying infrastructure and destination cloud endpoints.
- **Extensibility, customizations, and integrations:** Full extensibility and customization with Orchestrator, Action-Based Extensibility (ABX), and built-in integrations with popular third-party tools.
- **Configuration management:** Day 1 and 2 control for virtualized and cloud environments with intuitive configuration automation, vulnerability remediation, and compliance enforcement.
- **Infrastructure pipelining:** User-friendly release automation pipelines that enable continuous integration and continuous delivery (CI/CD) of infrastructure resources.

All things considered, a modern infrastructure automation solution can help reduce overall application development cost, complexity, and time to market. And it can optimize operations across a multi-cloud environment for IT efficiency, security, and agility that supports business revenue growth. The VMware Aria Automation solution can help transform existing manual workstreams associated with the application life cycle and inconsistent policy and tooling across different applications and environments. It can also help mitigate the risk of inconsistent security and compliance that could compromise the overall security of the data center.

IN THIS CHAPTER

- **Exploring public and private cloud accounts**
- **Diving into multi-cloud resources, cloud zones, resource mappings, and Aria Automation Templates**
- **Looking beyond Infrastructure as a Service**

Chapter 2
Meeting Assembler

Aria Automation Assembler is a cloud-based service that you use to create and deploy virtual machines, containers, applications, and services to your cloud infrastructure.

Assembler is VMware's approach to building a seamless, developer-relevant, Infrastructure as Code (IaC) first experience between multiple cloud endpoints.

At first glance, Assembler looks like the place where you can put things together like toy blocks, but there's a lot more going on. As a cloud administrator, you can:

- Configure the cloud vendor infrastructure to which your users deploy their Aria Automation Templates
- Set up projects to link the end users with the infrastructure resources
- Import templates and OVA files to support template developers using the marketplace
- Delegate the user management and deployment infrastructure to project managers, freeing you up to focus on your cloud resources

As an Aria Automation Template developer, you can:

- Create and iterate on templates until they meet your development needs
- Deploy templates to the supporting cloud vendors based on your project membership
- Manage the deployed resources throughout the development life cycle

TECHNICAL STUFF

Aria Automation Cloud has a monthly release cycle, meaning that newer features and enhancements are available every month.

Learning About Cloud Accounts: Public and Private

Assembler is designed to act as a conduit to consume services from multiple cloud environments, with public cloud treated as a first-class citizen within the platform. You most likely want to have multiple public cloud endpoints configured in your environment: AWS, Azure, GCP, and probably VMware Cloud on AWS, Azure, GCP, Oracle, Dell, and so on.

To do so in a truly agnostic resource provisioning and management way, you need to take the accounts and create a relationship between those endpoints. In Assembler, the main types of Cloud Accounts connections are AWS, GCP, Azure, NSX-T, NSX-V, vCenter Server, and VMware Cloud on AWS.

REMEMBER

vCenter Server is the centralized management utility for VMware and is used to manage virtual machines, multiple ESXi hosts, and all dependent components from a single centralized location.

TIP

Except for VMware Cloud on AWS, any VMware Cloud Stack (VMware Cloud on Azure, GCP, Oracle, Dell, and so on) can use the same vCenter and NSX-T Manager Cloud Account Connectors.

Once an endpoint is added, a discovery process is initiated to collect all the resources contained within each endpoint. Assembler collects information around the following object types: Compute, Networks, Security, Storage, Virtual Machines, and Volumes.

Aria Automation Cloud also can discover Kubernetes resources objects, such as Nodes, Namespaces, Resource Limits, and so on. These become available whenever you add existing Kubernetes endpoints or deploy a new cluster at the Resource Kubernetes Section.

Kubernetes is an open-source container orchestration system for automating software deployment, scaling, and management. Originally, Google designed Kubernetes, but now the Cloud Native Computing Foundation maintains the project.

Assembler was built from the ground up to answer customers' public cloud needs. But interacting with the private cloud is a critical component of most companies' multi-cloud strategy. How do you leverage on-premises resources with a platform hosted in Software as a Service (SaaS)?

Enter the Cloud Proxy, a virtual appliance that spins up a series of service proxies for interacting with on-premises services. Remember that there's a connector for NSX-T and vCenter Server; Assembler connects to them via Cloud Proxy.

Each proxy service running inside Cloud Proxy is a different container, making it extremely modular and easy to update. Even though it is a remote appliance, Cloud Proxy management operations, such as upgrades, restarts, and so on, can be executed from the Assembler UI itself.

Getting to Know Multi-Cloud Resources

Once you connect your multi-cloud environments to Aria Automation Assembler, it's time to organize how those resources are going to become available and consumed by your end users.

Cloud zones

At this point Assembler has discovered and collected all the resources contained within each endpoint, among them the compute resources. But because you may not want to share all those compute resources with customers, Assembler allows you to create *cloud zones* (shown in Figure 2-1), which are logical constructs containing compute resources available in regions/zones/clusters from any cloud, that you can organize and assign into zones (cloud zones).

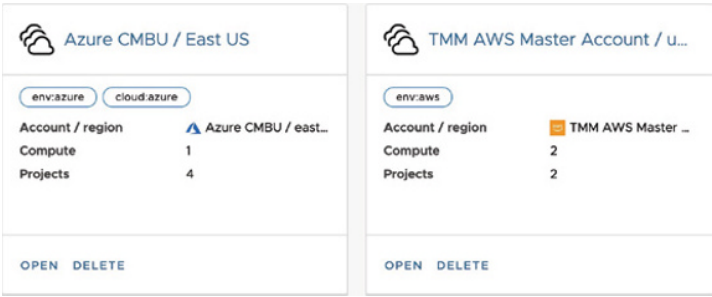

FIGURE 2-1: Cloud zones can be added and updated at any time to introduce more resources without disturbing existing projects.

Cloud zones are bound to a construct known as *projects*, to give users access to compute resources. Projects use various mappings and constraints' tag combinations to determine which definitions within the mappings to use.

Resource mappings and Aria Automation Templates

With the Aria Automation Templates in place, you can now create mapping relationships (OS Images and Virtual Machines Flavors) and Compute, Network, and Storage profiles.

Mappings and profiles describe VMware's multiple clouds' resources capabilities and encapsulates them behind a common abstraction definition, that can later be called a Cloud Template:

- **Flavor Mappings:** Equivalent to a sizing definition. "What is a small, medium, or large?"
- **Image Mappings:** Mapping of Multi-Cloud OS image name definitions under a single name reference.
- **Network Profiles:** Collections of network details. For on-premises constructs, this includes IP Ranges, Security Groups, subnets, and the like.
- **Storage Profiles:** Storage types; SSD versus standard disks, IOP limits, and so forth.

Each of these resources can have constraint tags applied to them. *Constraint tags* help the provisioning engine (consumed within Aria Automation Templates) decide which of the resource mappings to leverage.

If you want a workload to land on high-speed storage, you would tag a storage profile with `type:performance`. If you want to leverage an external IP address on a workload, you would tag a network as `network:external`. Each of these constraints can be referenced on the Aria Automation Templates and steer the workloads to land in/on a desired cloud location.

TIP

Constraint tags are different from regular metadata tags, which can also be used and defined in the Aria Automation Templates for all different types of resources.

Aria Automation Templates

The Aria Automation Templates can be as simple or as complex as you want them to be. There are multiple methodologies around creating these templates, including version control with Git-based version control servers. For now, we look at a sample Aria Automation Template, as shown in Figure 2-2.

As you work with Aria Automation Templates, here are a few key points to remember:

- On the far left you have a set of objects or constructs representing resources that you can use, which can then be dragged onto the canvas in the middle of the screen. You can take those objects and build connections/dependencies with them.
- As you add content, the YAML (Yet Another Markup Language) is populated in the panel on the right.
- In the YAML, you have inputs configured for username, password, and the number of deployments to create; you can define as many inputs as needed, and they're of different types, strings, integers, objects, and so on.
- In this example, we're using cloud agnostic objects, including a load balancer. This is all configured within the YAML to the right.
- The set of code under the cloudConfig section maps to Cloud-Init.

TECHNICAL STUFF

Cloud-Init is an industry standard cloud computing configuration tool that runs a set of scripts at the time of an instance startup. Think of it like the Configuration Spec in vCenter. This allows you to initialize the operating system, push configurations, commands, and packages into a resource you're building.

```
resources:
  frontend:
    type: Cloud.Machine
    dependsOn:
      - mysql
    properties:
      flavor: medium
      count: '${input.frontendcount}'
      image: ubuntu
      name: '${self.resourceName}'
      tags:
        - key: OC Deployment
          value: '${env.deploymentId}'
        - key: Application-Tier
          value: frontend
        - key: Application-Name
          value: '${env.deploymentName}'
      networks:
        - network: '${resource.Cloud_Network_1.id}'
      cloudConfig: |
        repo_update: true
        repo_upgrade: all
        apt_source:
          - source: deb http://archive.ubuntu.com/ubuntu main

        packages:
          - apache2
          - php
          - php-mysql
          - libapache2-mod-php
          - php-cli
          - php-common
          - php-intl
          - php-gd
          - php-mbstring
          - php-xml
          - php-zip
```

FIGURE 2-2: Aria Automation Template — IaC View (Abstract).

Going Beyond Infrastructure as a Service

What's not shown in Figure 2-2 is that your capabilities expand beyond traditional Infrastructure as a Service (IaaS) deployments. You also have the capability of consuming Cloud Native primitives from AWS, Azure, GCP, and other cloud services, such as Kubernetes, Terraform, Configuration Management solutions (Ansible / Puppet / Aria Automation Config) and Custom Resources.

REMEMBER

Aria Automation Config is a modern configuration management platform with the performance, speed, and agility that IT teams need to manage large, complex IT systems and improve efficiency at scale. See Chapter 6 for more on Config.

For example, with AWS RDS, users can consume native MySQL database capabilities without having to manage a full-sized SQL database (see Figure 2-3).

```
AWS_RDS_Instance_1:
  type: Cloud.Service.AWS.RDS.Cluster.Instance
  properties:
    region: us-west-1
    account: MainTMM AWS
    identifier_prefix: moad-engine-
    instance_class: db.r4.large
    cluster_identifier: '${resource.AWS_RDS_Cluster_1.id}'
    count: '1'
    publicly_accessible: false
    engine: aurora-mysql
    engine_version: 5.7.mysql_aurora.2.03.2
    tags:
      - key: Cost Center
        value: '${input.costCenter}'
AWS_RDS_Cluster_1:
  type: Cloud.Service.AWS.RDS.Cluster
  properties:
    region: us-west-1
    account: MainTMM AWS
    cluster_identifier_prefix: moad-cluster
    database_name: opencart
    master_username: ocuser
    master_password: '${input.password}'
    port: '3306'
    skip_final_snapshot: 'true'
    engine: aurora-mysql
    engine_version: 5.7.mysql_aurora.2.03.2
    vpc_security_group_ids:
      - sg-bf0c04dd
    tags:
      - key: Cost Center
        value: '${input.costCenter}'
```

FIGURE 2-3: Aria Automation Template — AWS RDS Resources (Abstract).

But what happens after you deploy workloads?

Diving into Deployments and Beyond

The deployments in Assembler are the Aria Automation Templates that are provisioned on your cloud account platforms. A successfully deployed cloud template represents your final goal as an administrator or cloud template designer. A deployment enables you to manage the life cycle of workloads via Day 2 operations as a whole deployment and/or per individual resources.

Let's say you want to add more constructs to the existing deployment. Perhaps you were doing functional testing and now you'd like to run stress testing, which requires more servers. Continuing with the cloud template example, can you add a new tier for the front end? No problem! You can simply modify the cloud template by selecting the Update an Existing Deployment option to push those changes in.

IN THIS CHAPTER

- Finding out what Service Broker is
- Reviewing content source types, policies, and deployments

Chapter 3
At Your Service: Presenting Service Broker

Service Broker provides a user-accessible service catalog of items that can be requested and managed by an end user. The end user can view the life cycle of their deployments and perform Day 2 actions, as well as monitor the deployment progress.

The cloud administrator can use Service Broker to manage and present simplified catalog items, customize forms, define which catalog items are available to which users, and define policies to manage resource use, deployment lease times, Day 2 actions, and approval policies.

Exploring the Service Broker Catalog

The catalog acts as a type of storefront where resources can be requested and later consumed by end users. Content is displayed as requestable items, as shown in Figure 3-1. The rest of this chapter takes you on a quick tour through the items that are most often requested.

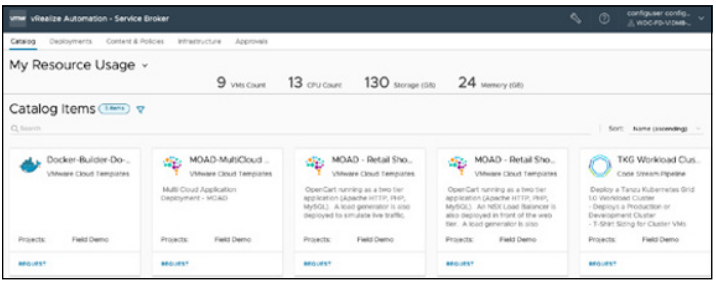

FIGURE 3-1: The Service Broker Catalog.

Getting to Know Content and Policies

The Content and Policies tab enables Service Broker administrators to configure the Service Broker Catalog and deployment controls, as follows:

- **Content Sources:** Here's where you configure the sources for catalog items, which we discuss in the next section.

  ```
  https://learnservicebroker.github.io/Content-
    and-Policies/Content-Sources/
  ```

- **Content Sharing:** Go here to configure which catalog items are available to which projects.

  ```
  https://learnservicebroker.github.io/Content-
    and-Policies/Content-Sharing/
  ```

- **Content:** Check out this section to view and configure all the configured catalog items and custom forms.

```
https://learnservicebroker.github.io/Content-
  and-Policies/Content/
```

- **Policies:** Here's where you define and enforce policies for catalog items. We discuss this a bit later in this chapter.

```
https://learnservicebroker.github.io/Content-
  and-Policies/Policies/
```

Learning About Content Source Types

From the Content and Policies tab, click Content Source types to see which types can be configured (Figure 3-2):

- **Extensibility Actions**
- **Assembler Action Based Extensibility (ABX) actions as Catalog Items:**

```
https://learncloudassembly.github.io/
  Extensibility/Library/Actions/
```

- **Aria Orchestrator Workflow:** Configure Aria Orchestrator Workflows as Catalog Items
- **AWS CloudFormation Templates**
- **Marketplace VM Templates (OVA)**
- **Pipelines**
- **VMware Aria Automation Templates**
- **vRA 7.x Catalog Items**
- **VMware Aria Automation 8.x Template (AA Cloud only):** Configure VMware Aria Automation 8.x on-premises Aria Automation Templates as a Catalog Item

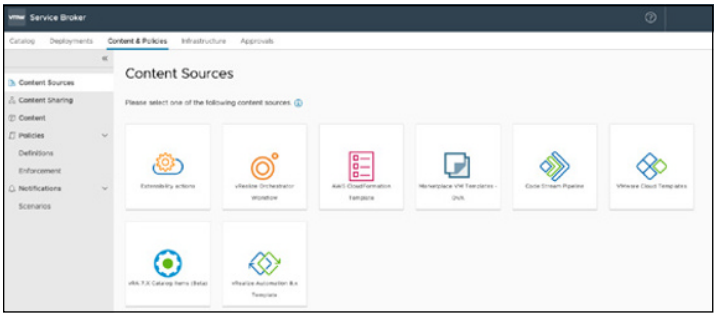

FIGURE 3-2: Content Sources.

Perusing Policies

Policies allow administrators to control approvals for resource provisioning requests, resource use, Day 2 actions, and life cycle. There are five types of policies that can be created under the Definitions page, which we discuss separately in the following sections:

- Resource quota policy
- Deployment limit policy
- Lease policy
- Day 2 actions policy
- Approval policy

Resource quota policy

This policy restricts total resource use across organizations, projects, and users by providing the ability to apply numerical limits for each resource type. Limits can be applied on the following resource types for each entity: CPU count, VM count, memory limits, and storage limits.

Figure 3-3 shows an example resource quota policy; in it, a limit is set for both the Test and Development Projects. Each project has a total limit of 12 CPU, 128 GB of RAM, and 1,000 GB of storage. Each user in either project can deploy up to 8 VMs in total.

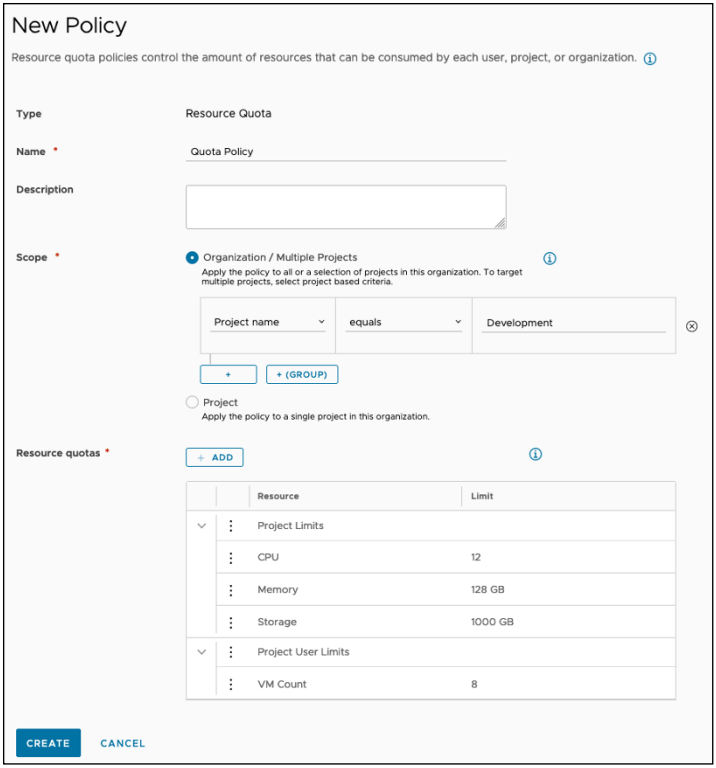

FIGURE 3-3: An example resource quota policy.

Deployment limit policy

Deployment limit policies control the amount of resources that can be used by deployments. The scope is also at the Organization and Project; however, limits are applied within deployments. Limits are applied on the following resources: CPU count, VM count, and memory limits.

Figure 3-4 shows an example deployment limit policy in which a limit is set for the Test Project. The project has a total limit of 6 CPUs and 24 GB RAM, and each user in the project is limited on a per deployment basis as well.

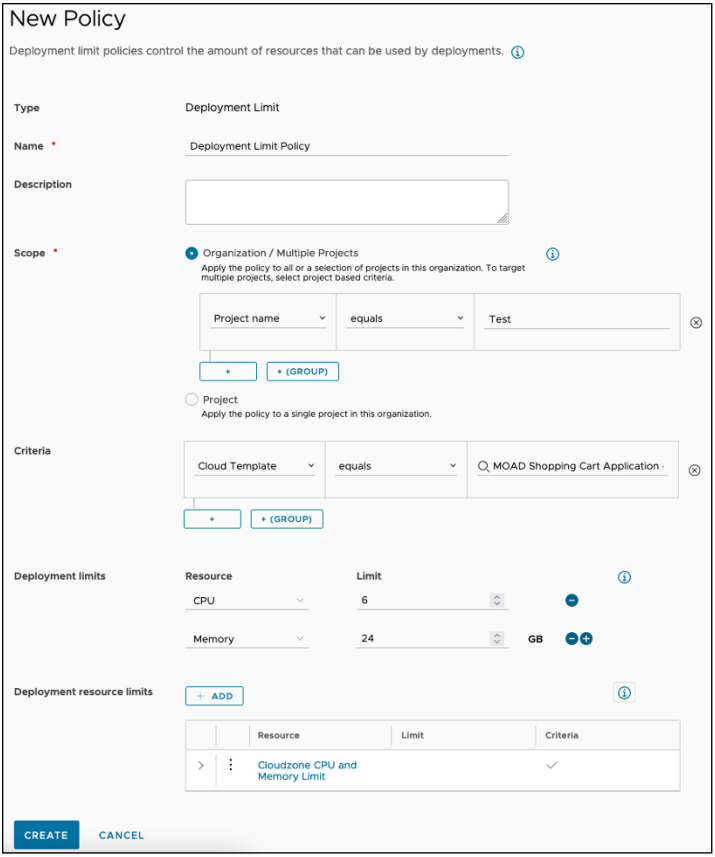

FIGURE 3-4: An example deployment limit policy.

Lease policy

Lease policies control how long deployments are available for use before they're destroyed and the resources are reclaimed:

- **Enforcement type (Hard or Soft):** In the event of conflicting policies, a Hard Policy overrides a conflicting Soft Policy.
- **Maximum Lease (days):** This is the number of days a deployment can be leased for, and the number of days a deployment's lease can be renewed for.
- **Maximum Total Lease (days):** This is the maximum number of days a deployment can be leased for, including lease renewals and the *Grace period* (see next bullet).

- **Grace period (days):** The number of days after a deployment lease has expired that the deployment will be destroyed.

In the example lease policy, shown in Figure 3-5, any deployment in the Production project will have a maximum lease of 30 days, with a total lease time of 100 days and grace period of 5 days. Deployments will be destroyed at the end of the grace period once renewals are exhausted or if available renewals aren't used.

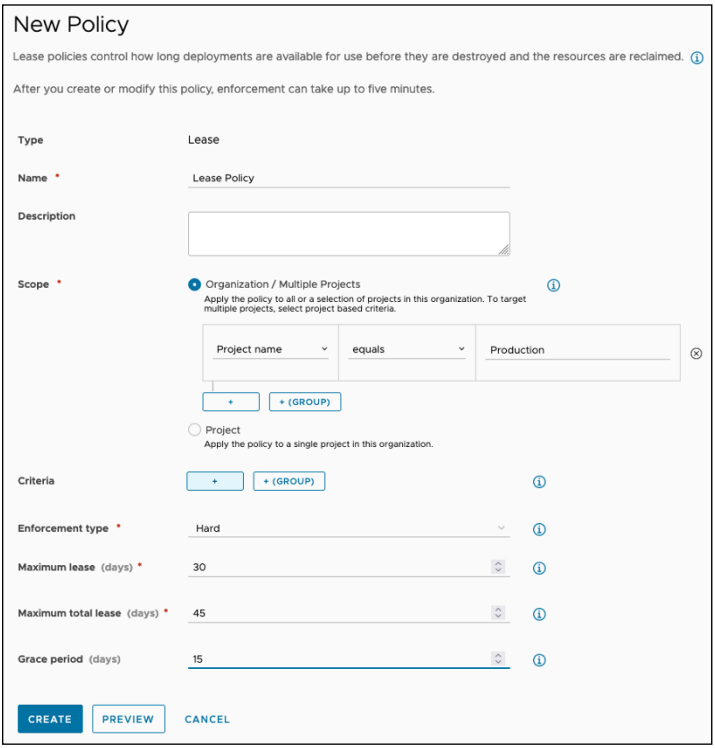

FIGURE 3-5: Lease policy creation.

Day 2 actions policy

Day 2 action policies control Day 2 actions that users will have access to for deployed resources, including:

- **Role:** You can select the Administrator, Member, or a custom role that the policy will apply to.

- **Actions:** Select one or more out-of-the-box Day 2 actions, or Custom Day 2 actions to enable this policy.

In the example shown in Figure 3-6, Day 2 actions, including power actions and disk management, are set for vSphere objects deployed using the selected Aria Automation Template criteria.

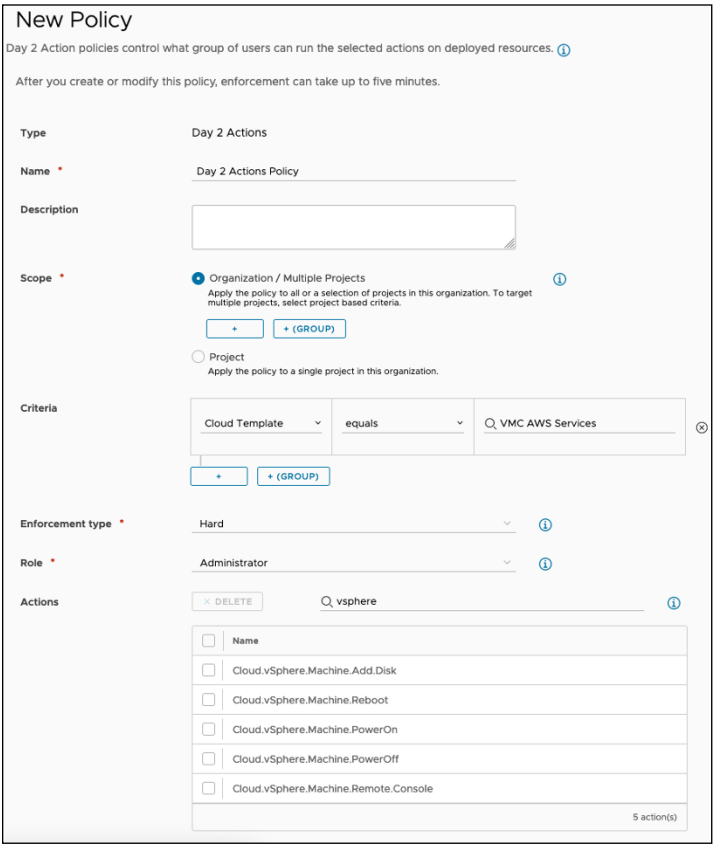

FIGURE 3-6: Day 2 policy creation.

Approval policy

Approval policies control who must agree to a deployment or Day 2 action before the request is provisioned. These are the options you have for setting approval policies:

- **Approver mode:** *Any* allows any user from the Approvers group to approve the request; *All* requires all of the users in the Approvers group to approve the request.
- **Approvers:** This is a list of users or groups that can approve the request.
- **Auto expiry decision:** The request is automatically approved or rejected if the *Auto expiry trigger* is reached without an *Approver* responding.
- **Auto expiry trigger:** You can set the number of days (up to 7) that Approvers have to respond before the Auto expiry decision takes effect.
- **Actions:** This is a list of actions that the approval policy will apply to.

Figure 3-7 shows an example in which the selected user must approve all requests for the specified Cloud Template in Project Production and for the listed custom action.

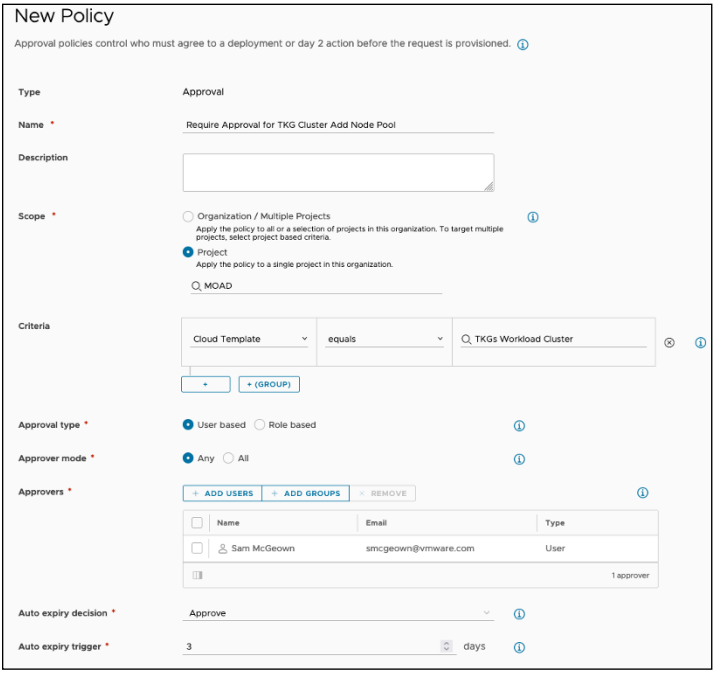

FIGURE 3-7: Approval policy creation.

Diving into Deployments

The Deployments tab provides Service Broker users with visibility into provisioned Deployments and Resources that they have permission to view or manage. This is the same view that's available in Assembler. (See Chapter 2 for more about Assembler.)

The Resources tab allows you to view and manage the resources provisioned as part of your deployments; you can see more information for a deployment by selecting the deployment name.

IN THIS CHAPTER

- Finding out what Pipelines is
- Configuring endpoints and variables
- Understanding and creating Pipelines
- Managing executions and triggers
- Reviewing Pipeline dashboards

Chapter 4
Laying Down Pipelines

Pipelines is a continuous integration and delivery (CI/CD) release pipeline tool provided as part of Aria Automation (either as SaaS through VMware Cloud Services or an on-premises deployment) that enables developers to model and automate the entire release process. It incorporates a release dashboard to help you keep track of all the various release KPIs, and it acts as the glue between all existing DevOps tools in the release process.

Configuring Pipelines

Pipelines has native integrations with a range of Endpoints and can help teams to deliver software and code changes faster, more reliably, and with higher quality while reducing manual operations and the operational risk that's traditionally associated with releases. Pipelines can also be extended using the Custom Integrations feature to interact with almost any third-party system that has an API or CLI.

Endpoints

Endpoints allow Pipelines to connect to remote applications and data sources. Most Pipeline tasks leverage the endpoints to execute their actions.

Pipelines includes the following endpoint types:

- **Bamboo:** Provides configuration for Bamboo Tasks.
- **Docker (Host):** Provides an execution environment for CI Tasks and Custom Integrations.
- **Docker Registry:** Provides credentials for accessing a specific Registry in the context of a Pipeline Workspace.
- **Email:** Provides configuration for sending Pipeline Notifications, Task Notifications, and User Operations.
- **Gerrit:** Provides configuration for use in Gerrit.
- **Git:** Provides configuration for Git Repository Triggers.
- **Jenkins:** Provides integration to a Jenkins server.
- **Jira:** Provides configuration for sending Pipeline Notifications and Task Notifications.
- **Kubernetes:** Provides configuration for a Kubernetes cluster to use with the Kubernetes Task.
- **TFS:** Provides integration to a TFS server and allows the use of a TFS Task.
- **VMware Aria Automation Orchestrator:** Provides integration to an Aria Orchestrator server for use in tasks. (See Chapter 5 for more about Orchestrator.)

Variables

Variables are a great way to keep reusable text values or secrets for use in Pipelines in one central place. Variables can be used to provide secure access to credentials or configuration information. Using Variables ensures that sensitive information isn't exported if you need to export your pipelines, and allows you to control access to that sensitive information.

Variables can be one of the following types:

- **Regular:** Value is plain text.
- **Secret:** Value is hidden but can be used in Pipelines.
- **Restricted:** Value is hidden and can only be accessed in Pipelines by administrators.

TIP

Variables are accessed in Pipelines by typing the dollar symbol ($), which opens the Pipeline Variables menu. A Variable with the name of `mysecret` will be accessed using the notation `${var.mysecret}` in Pipelines and Endpoints.

Custom Integrations

Custom Integrations allow you to write reusable custom code in Python, Shell, or NodeJS, and execute your code as a Custom Task in a stage of a Pipeline. When the Custom Integration task is executed, it uses the docker host or Kubernetes endpoint and container image for the parent Pipeline.

Creating Pipelines

A Pipeline is the primary mechanism for sequencing all the tasks that need to be performed. It's composed of one or more stages (such as Testing or Release), with one or more tasks (such as "Test Build" or "Deploy Blueprint") in each stage.

Pipeline settings

The Pipeline tab, shown in Figure 4-1, allows you to set the pipeline name, execution concurrency, description, icon, and tags.

FIGURE 4-1: Enter settings information on the Pipeline tab.

The Workspace tab configures the environment in which the pipeline runs:

- **Type:** Select whether to execute Pipeline Tasks on a Kubernetes or Docker endpoint.
- **Builder image URL:** Configures the container image that will be used for CI tasks or Custom Integrations.
- **Image Registry:** The Docker Registry endpoint to use to pull the Builder image.
- **Namespace (Kubernetes only):** Specifies a Kubernetes Namespace in which the Kubernetes Deployment running the container image will be created. If the Namespace doesn't already exist, then Pipelines will automatically create it.
- **Proxy type (Kubernetes only):** Pipelines communicates tasks with the CI Agent running on the container image Pod via a NodePort on the Kubernetes Worker, or a Load Balancer (recommended).
- **NodePort (Kubernetes only):** If you select the NodePort proxy type, you can leave this value blank to use an ephemeral port number, or specify a port between 30000 and 32767.
- **Persistent Volume Claim (Kubernetes only):** Stores the logs and output of the CI Agent running on the container image. If you don't specify a Persistent Volume Claim, then an ephemeral volume type will be used.
- **Working directory:** This is a directory within a container image that's used when running commands.
- **Cache:** This is accessible to each pipeline run and can be used to cache files and folders that are common between pipeline runs.
- **Environment Variables:** Used to pass environment variables to a container.
- **CPU and memory limit:** If a CI task requires significant resources, the container's allocated CPU can be increased.
- **Git clone:** If the pipeline is triggered by a Git webhook, CI tasks will automatically clone the Git repository.

Pipeline inputs and outputs

Input parameters are passed to the Pipeline before execution. Output parameters are produced by the Pipeline as a result of its

actions. You can nest pipelines using the Pipeline task to return the results to the parent pipeline.

You can also auto-inject parameters for Gerrit, Git, or Docker variables when you're using Triggers.

The Pipeline model

The Model tab is where you configure the stages and tasks of the pipeline; it's where you spend most of your time when creating and editing pipelines (see Figure 4-2).

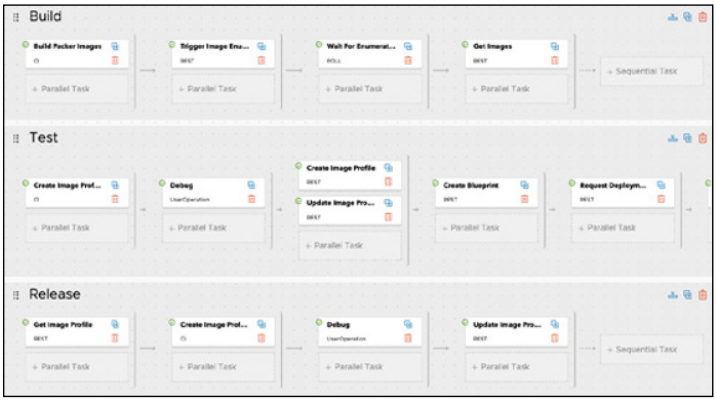

FIGURE 4-2: The Model tab.

Stages

Pipeline stages are logical groupings of tasks to reflect the structure of the process; for example, your process has a Build, Test, and Release phase. The Pipeline stages can be configured to reflect this.

Tasks

Pipelines Tasks are the basic units of a Pipeline, with different task types interacting with different endpoints or systems.

TIP

Some configuration is common across all task types:

- **Precondition:** The precondition field can be used to determine if a task should be executed.
- **Continue on failure:** If checked, the failure of this task will not cause the entire Pipeline to fail.

- **Task Notifications:** Task notifications are almost identical to Pipeline notifications except that they offer a specific event for the task.
- **Rollback:** The Task Rollback setting allows you to configure a Pipeline that will be executed if the task fails.

Specific configuration for each task is required, depending on the task type selected. Available task types are:

- **Bamboo:** Execute a Bamboo Plan on a specific Bamboo Endpoint.
- **CI:** Enables almost any action in your Pipeline by executing the CI task script in the context of a running container. It's an incredibly powerful and flexible task type because the image can have almost any tool or program in it.
- **Condition:** Can be used to evaluate the success of previous stages before moving on with the Pipeline.
- **Custom:** The Custom Task allows you to use Custom Integrations in Pipeline Stages.
- **Jenkins:** Execute a Jenkins Task on a Jenkins Endpoint.
- **Kubernetes:** Allows for the execution of Get, Create, Apply, Delete, and Rollback actions against a Kubernetes Endpoint.
- **Pipeline:** Allows you to nest existing Pipelines within a parent pipeline.
- **Poll:** Most commonly used in combination with the REST task to Poll for a job completion status; however, it can be used to Poll any HTTP server.
- **PowerShell:** Allows you to execute PowerShell scripts on a remote PowerShell server using PSRemoting.
- **REST:** A powerful and adaptable client that lets you interact with any standard REST API.
- **SSH:** This task type allows you to execute code on a remote machine — in this case, over SSH.
- **User Operation:** Provides a way to include approvals within a Pipeline Execution.
- **Aria Automation Template:** Can be used to create, update, delete, and rollback deployments in Aria Automation Assembler.
- **VMware Aria Automation Orchestrator:** Allows you to execute an Orchestrator Workflow.

Using variables in Pipelines

Most configurable fields within a Pipeline can also use Variables, references to Input parameters, and the output of other Pipeline tasks or Pipeline properties by using a reference. These can be accessed by typing the dollar sign ($), which brings up the auto-completion.

Pipeline notifications

The Notifications tab allows you to configure notifications for pipeline events (completion, waiting for user interaction, failure, cancellation, and starting) using an email endpoint, a Jira endpoint, or by creating a webhook with a POST, PUT, or PATCH payload.

Executing Pipelines

Pipelines can be executed directly from the Pipeline editor, the Pipeline page, the Executions page, or from Aria Automation Service Broker. Executing a Pipeline prompts the user for any comments and inputs configured for the Pipeline.

The Executions page (see Figure 4-3) provides a detailed account of every Pipeline Execution that can be filtered by Pipeline, Status, Tag, or any other pipeline property. It provides an at-a-glance view of which pipelines have failed, where they've failed, and the error messages returned.

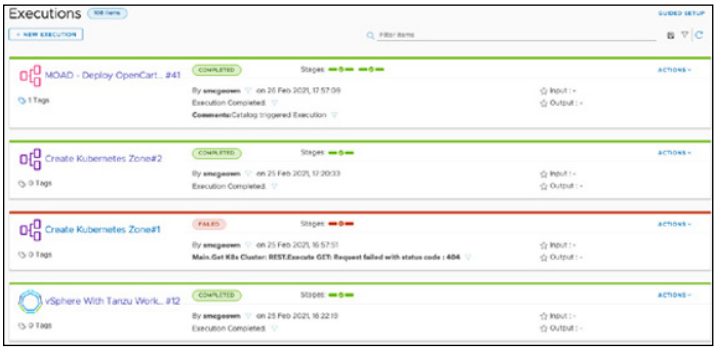

FIGURE 4-3: The Executions page.

Performing User Operations

User Operations provide a way to include approvals within a Pipeline Execution using the User Operation Task. The User Operations page provides a dashboard of all active and inactive user operations that the logged-on user is named for in the Approvers list.

Users with administrative rights in Pipelines can view, approve, or reject user operations for which they aren't in the Approvers lists.

Integrating with Pipeline Triggers

Triggers are a way for Pipelines to integrate with Docker, Gerrit, and Git life cycles. Pipelines connects to the respective endpoint through a webhook.

A *webhook* is configured by an administrator, or through Pipelines, for a push or pull request event on the Triggers tab. Through the webhook, any code change events on the remote repository are received by the trigger in Pipelines.

Looking at Dashboards

Pipelines users can view Dashboards to review historic data for all Pipeline executions.

In addition to the automatically generated default dashboards, developers and administrators can create custom dashboards to view specific results by adding widgets from the menu to display statistics.

IN THIS CHAPTER

- Perusing the platform from architecture to plug-ins and key features
- Discovering workflows, actions, and policies
- Understanding administration, inventories, and assets

Chapter 5
Conducting with Orchestrator

VMware Aria Automation Orchestrator is a development- and process-automation platform that provides a library of extensible workflows to allow you to create and run automated, configurable processes to manage VMware products as well as other third-party technologies.

Orchestrator automates management and operational tasks of both VMware and third-party applications, such as service desks, change management systems, and IT asset management systems.

Platform: From Architecture to Plug-Ins

This section dives into the Aria Automation Orchestrator core components and the importance of the plug-ins to enable features.

Plug-ins

The Aria Automation Orchestrator plug-in architecture allows you to access and control external technologies and applications. Plug-ins extend the Orchestrator scripting engine with new object types and methods.

The external technologies that you access by using plug-ins include virtualization management tools, email systems, databases, directory services, and remote-control interfaces. Orchestrator provides a standard set of preinstalled plug-ins, and you can develop custom plug-ins to access other applications.

Architecture

Aria Automation Orchestrator is composed of three distinct layers:

- An orchestration platform that provides the common features required for an orchestration tool
- A plug-in architecture to integrate control of subsystems
- A library of workflows

These layers enable Orchestrator to provide high availability and scale capacity along with the growth of your cloud. It's worth mentioning that with each deployed Orchestrator appliance a preconfigured PostgreSQL Database is included to store data.


TECHNICAL STUFF

Orchestrator has become a unified integration layer of Aria Automation and is also helping vSphere vCenter and Aria Operations to build the Self-Healing Datacenter concept when Aria Automation Orchestrator is handling the automatic remediation process.

Key features

Aria Automation Orchestrator includes several key features:

- **Persistence:** A production-grade PostgreSQL database is used to store relevant information, such as processes, workflow states, and the Orchestrator configuration.
- **Central management:** The application server-based platform, with full version history, can store scripts and process-related primitives in the same storage location. This way, you can avoid scripts without versioning and gain proper change control on your servers.
- **Check-pointing:** Every step of a workflow is saved in the database, which prevents data-loss if you must restart the server. This feature is especially useful for long-running processes.
- **Control Center:** Control Center is a web-based portal that increases the administrative efficiency of Orchestrator

instances by providing a centralized administrative interface for runtime operations, workflow monitoring, and correlation between the workflow runs and system resources.

- **Versioning:** All Orchestrator platform objects have an associated version history. Version history is useful for basic change management when distributing processes to project stages or locations.
- **Git integration:** You can integrate a Git repository to further improve version and source control of your Orchestrator content. With Git, you can manage workflow development across multiple Orchestrator instances.
- **Scripting engine:** The scripting engine is enhanced with basic version control, variable type checking, name space management, and exception handling.
- **Workflow engine:** The workflow engine allows you to automate business processes. It uses objects to create a step-by-step process automation in workflows.
- **Policy engine:** You can use the policy engine to monitor and generate events to react to changing conditions in the Aria Automation Orchestrator Client server or a plugged-in technology. Policies can aggregate events from the platform or the plug-ins, which helps you to handle changing conditions on any of the integrated technologies.
- **Aria Automation Orchestrator Client:** Create, run, edit, and monitor workflows with the Aria Automation Orchestrator Client. You can also use the Aria Automation Orchestrator Client to manage action, configuration, policy, and resource elements.
- **Development and resources:** The Aria Automation Orchestrator landing page provides quick access to resources to help you develop your own plug-ins for use in Orchestrator.
- **Security:** Orchestrator provides advanced security functions, such as Public Key Infrastructure (PKI), Digital Rights Management (DRM), Transport Layer Security (TLS), and Access Rights Management (ARM), to provide control over access to processes and the objects manipulated by these processes.
- **Encryption:** Orchestrator uses a FIPS-compliant Advanced Encryption Standard (AES) with a 256-bit cipher key for encryption of strings.

REMEMBER

Aria Automation Orchestrator is an open platform that can be extended with new plug-ins and content and can be integrated into larger architectures through a REST API.

Diving into the Dashboard

The Aria Automation Orchestrator Client dashboard, available as a web-based UI, provides a useful tool for monitoring, managing, and troubleshooting Aria Automation Orchestrator System and Client workflows. Information on the Aria Automation Orchestrator Client dashboard is spread among five panels:

- The **Workflow Runs** panel provides visual data about the number of running, waiting, and failed workflow runs.
- The **Favorite Workflows** panel displays workflows added to favorites.
- The **Waiting for Input** panel is where you see pending workflow runs that require further user interaction. These workflows are also displayed in the notifications menu in the upper-right corner of the UI.
- Manage your recent workflow runs in the **Recent Workflow** runs panel, which shows the name, state, start date, and end date of the workflow run.
- The **Requiring Attention** panel displays failed workflow runs and workflow run performance metrics.

Additionally, you can use the Aria Automation Orchestrator Client Dashboard and profiling feature to gather useful metrics about your Orchestrator environment.

Looking at Workflows, Actions, and Policies

You develop workflows and actions in the Aria Automation Orchestrator Client. Workflow and actions development involves using the workflow editor, the Aria Automation Orchestrator APIs, and the JavaScript, Python, Node.js, and PowerShell scripting languages.

Policies can be defined for aggregating events from the platform or the plug-ins, which helps you to handle changing conditions on any of the integrated technologies.

Workflows

Orchestrator provides a standard library of workflows, actions, and policies that you can use to automate operations in your virtual infrastructure.

REMEMBER

The workflows in the standard library are locked in a read-only state.

Workflows combine actions, decisions, and results that, when performed in a particular order, finish a specific task or a specific process in a virtual environment. Workflows perform tasks such as provisioning virtual machines, backing up, performing regular maintenance, sending emails, performing SSH operations, managing the physical infrastructure, and other general utility operations.

Workflows accept inputs according to their function. You can create workflows that run according to defined schedules, or that run if certain anticipated events occur. Information can be provided by you, by other users, by another workflow or action, or by an external process such as a web service call from an application. Workflows perform some validation and filtering of information before they run.

TIP

Workflows can call upon other workflows. For example, you can have workflow that calls up another workflow to create a new virtual machine.

REMEMBER

You create workflows by using the Aria Automation Orchestrator Client interface's integrated development environment (IDE), that provides access to the workflow library and the ability to run workflows on the workflow engine.

Actions

The Aria Automation Orchestrator Client provides libraries of predefined actions and an action editor for custom action scripts. *Actions* represent individual functions that you use as building blocks in workflows.

Actions are JavaScript, Python, Node.js, or PowerShell functions. Actions can take multiple input parameters and have a single return value. Actions can call on any object in the Aria Automation Orchestrator API Explorer, or objects in any API that you import into Orchestrator by using a plug-in.

Policies

Policies are event triggers that monitor the activity of the system. Policies respond to predefined events issued by changes in the status or performance of specific Orchestrator objects.

Policies are a series of rules, gauges, thresholds, and event filters that run certain workflows or scripts when specific predefined events occur in Orchestrator or in the technologies that Orchestrator accesses through plug-ins.


TIP

Orchestrator constantly evaluates the policy rules while the policy is running. For instance, you can implement policy gauges and thresholds that monitor the behavior of vCenter Server objects of the `VC:HostSystem` and `VC:VirtualMachine` types.

Keeping Track of Inventory

As an administrator, you can use groups to set what Orchestrator content users can view, such as accessing the inventory, managing Git repositories and full push/pull Git history, and reviewing the audit logs.


REMEMBER

As an administrator or user, you can use Aria Automation Orchestrator Client UI to access the Aria Automation Orchestrator content, including the object inventory.

The Technologies plug-ins (for example, vCenter, vRA, AD, and so on) expose all objects of the connected vCenter Server instances in the Inventory view.

IN THIS CHAPTER

- Studying Config
- Exploring the core features of Config
- Learning Config architecture

Chapter 6
Connecting with Config

Config is a modern configuration management and orchestration tool designed to help organizations manage their IT infrastructure. It sits on top of Salt and provides a UI into your Salt Infrastructure to help manage jobs and other activities. Config can be used to enforce configurations across applications and infrastructure using simple scripting and programming languages.

Config also uses event-driven automation to detect and auto-remediate drift. The software supports both agent and agentless options for managing systems; the agent provides some additional capabilities like self-healing.


TECHNICAL STUFF

Salt is a Python-based open-source remote execution framework for configuration management, automation, and orchestration. Salt supports the Infrastructure as Code (IaC) approach to deployment and data center management. Running commands on systems is the core function of Salt. A powerful advantage of using a platform like Salt is that it maintains the configuration of those systems.

Config provides a GUI to manage the Salt environment as well as features like:

- Role-based access controls
- Job scheduling
- Activity logging
- And much more!

Getting a General Overview of Config Architecture

Before we dive into the features of Config, it's helpful to understand the architecture of Salt and Config.

Aria Suite Lifecycle can be used to manage and deploy the Config appliance. The architecture is shown in Figure 6-1.

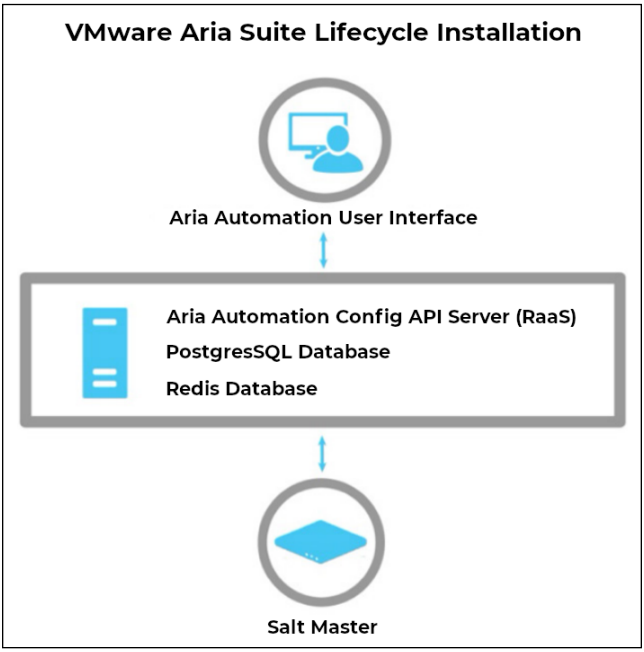

FIGURE 6-1: Single Node Config.

In the figure, Aria Suite Lifecycle provides management and installation capabilities for Aria Automation and Config. Notice that the PostgreSQL, Redis, and Config services are all in one appliance.

Looking at Multi-Node Architecture

For very large environments, certain architectures can provide more scalability and redundancy.

Figure 6-2 shows Config providing a user interface to the Salt Masters and Minions. Also notice that a PostgreSQL and Redis database are separated onto their own machine. Config stores all of its data in the PostgreSQL. The Salt Master is the main connection for Config and the rest of the nodes in the environment.

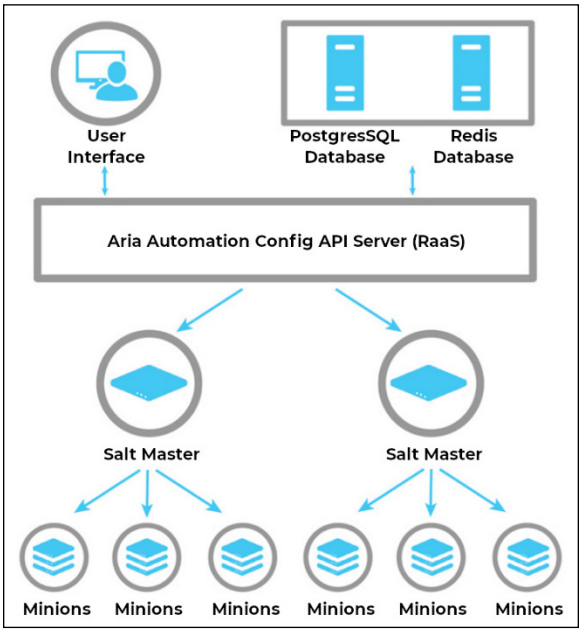

FIGURE 6-2: Config Architecture.

Each *Salt Master* (a server running all the services needed to push jobs and commands to the Minions) communicates with Config and provides visibility into the *Minions* (any system under management, like a webserver, database server, IIS server, and so on). Salt Masters and Minions communicate via a secure channel. AES and RSA keys are created at the time that the Minion is registered with the Salt Master. Then users can take advantage of a UI-based system for running jobs and commands against those Minions.

Figure 6-3 shows a Salt Minion attempting to register with a Salt Master. The Administrator can then accept the key manually, or the system can be configured to auto-accept keys based on certain criteria of the Minion, such as its MinionID. For instance, if the MinionID contains the word "prod" then logic can be written to automatically accept the key without Administrator intervention.

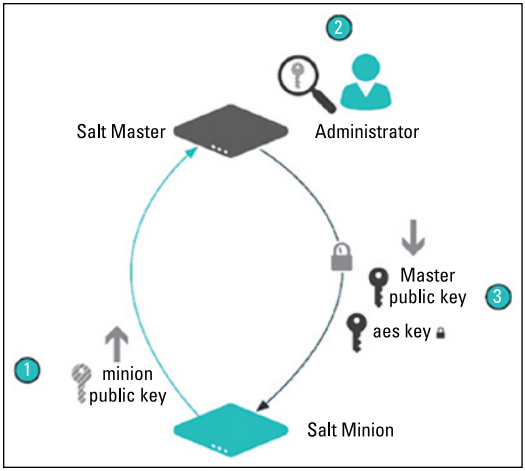

FIGURE 6-3: A Salt Minion registers with the Salt Master.

After the key has been accepted, the Administrator can begin to run jobs against the Minion.

REMEMBER

Salt Masters don't initiate communication with the Minions when a job is ready. The Minions are constantly asking the Salt Master whether it has a job it needs to run. This is done via the event-bus. Keep in mind that this works when the agent is installed.

Regardless of the architecture chosen, all of the features that are in the next section will be available.

Exploring Config Features

In this section we explain core Salt concepts and event-driven systems.

Salt grains

Salt comes with an interface to derive information about the underlying system. This is called the *grains interface* because it presents Salt with grains of information. Grains are collected for the operating system, domain name, IP address, kernel, OS Type, memory, and many other properties of the system.

A number of grains get set out of the box; however, custom grains can also be created. Think of grains as characteristics of the Minion

that can be presented to the Salt Master. The Salt Master can then make decisions based on the grains, such as whether a job should run on the Minion (targeting), set conditional commands that may depend on a certain type of grain, provide reporting. Figure 6-4 shows how grains are sent up to the Salt Master.

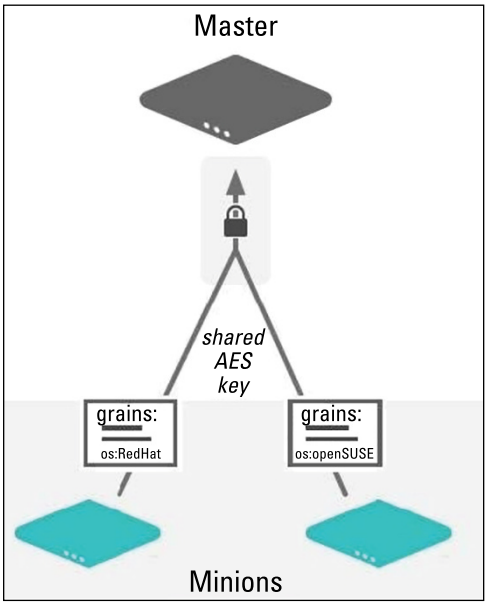

FIGURE 6-4: The grains are sent up to the Salt Master.

Salt states

Salt provides a method for configuring Minions by declaring which state a Minion should be in, otherwise referred to as Salt states. *Salt states* make configuration management possible. You can use Salt states to deploy and manage infrastructure with simple YAML files. Using states, you can automate recursive and predictable tasks by queueing jobs for Salt to implement without needing user input. Salt states can contain:

- **Multi-language renderers:** You can write states in languages like YAML, JSON, or Python.
- **Conditions and variable state options:** You can define certain conditions to be met before running a command in a state.

- **Inclusion of other states:** You can run a state file from another state file.

State files declare how you want the machine to be configured. A simple state file to install Apache would look something like Figure 6-5.

```
install_apache:
  pkg.installed:
    - name: apache2
```

FIGURE 6-5: Apache Install State.

The first line is just a custom identifier that explains what you want to do in the state; the `pkg.installed` is the function and module being called. The `name` parameter tells the system which package needs to be installed: In this case, it's `apache`. If you wanted to ensure that the Apache services is running, you would add more to this state file.

Pillar data

Salt pillar brings data into the cluster from the opposite direction to grains. While grains data is generated from the Minion, the pillar is data generated from the Salt Master. Pillars are organized similarly to states and act to coordinate pillar data to environments and Minions with access to the data. Pillar data is used for the following types of data:

- **Highly sensitive data:** Information transferred via the pillar has a dictionary generated for the targeted Minion and is encrypted with that Minion's key for secure data transfer.
- **Minion configurations:** Variables and other data can be stored in the pillar and used over and over again.

As shown in Figure 6-6, pillar data can be stored in a couple of places in the Salt environment, either on the Salt Master itself or in the Pillars section of the UI.

You can call the data from the pillar in a state file via a command `pillar.get` (see Figure 6-7). You can place the value of that pillar data into a variable and use it in your state files. Then your states can call the variables where needed.

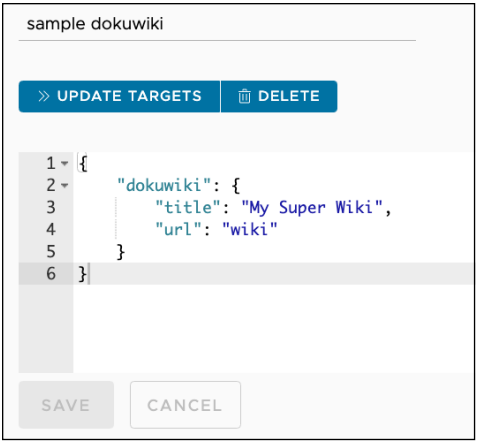

FIGURE 6-6: Pillar data store in Config.

```
{% set dokuwiki_url = salt['pillar.get']('dokuwiki:url', 'wiki') %}
{% set wiki_title = salt['pillar.get']('dokuwiki:title', 'My Wiki') %}
```

FIGURE 6-7: Calling pillar data from state.

Event bus-driven automation

The Salt event bus system is used to fire off events that can be seen and acted upon by both Salt Masters and Minions. Events are fired on the Salt Master for situations like Minion authenticating, job activity, Salt-key events, Minions starting up, and more. When these events fire they can be viewed by the administrator by running this command on the Salt Master:

```
salt-run state.event pretty=True
```

The events can also trigger things to happen in the environment without administrator intervention, like the auto-acceptance of a Minion key.

Reactors

The reactor system leverages the Salt Master event bus to trigger Salt state responses to targeted events (see Figure 6-8). Reactors expand Salt's ability with automation responses using pre-written remediation states. Reactors can be applied to a variety of scenarios:

- Restarting failed applications
- Automatic rollback

- Infrastructure scaling
- Notifying administrators
- Kicking off orchestration

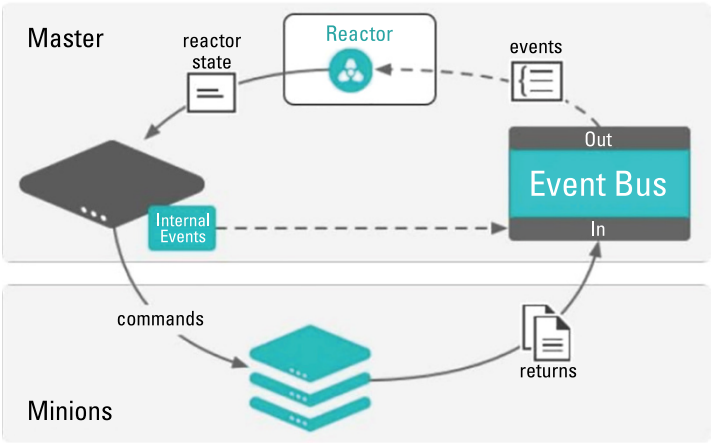

FIGURE 6-8: The reactor system leverages the Salt Master event bus to trigger Salt state responses.

Reactor configurations need to be on the Salt Master. Typical best practice is to create a `reactor.conf` file and copy it to the master in the directory `/etc/salt/master.d/reactor.conf`.

In the `reactor.conf` example, whenever a Minion tries to authenticate with the Salt Master (salt/auth), the reactor sees that on the event bus, and the reactor tells the Master to run the `accept-key.sls`. The `accept-key.sls` has instructions on what to do next.

Beacons

Beacons is a monitoring tool set up on the Minion. Beacons can be leveraged for a number of purposes:

- Check for file system changes
- Check for service status (service stopped, and so on)
- Shell activity, such as user logins
- Resource monitoring for network and disk usage

IN THIS CHAPTER

- Introducing the five strategic use cases for your business
- Highlighting outcomes and benefits
- Getting started: Your automation journey begins now

Chapter 7
Looking at Use Cases

A modern infrastructure automation solution can help reduce overall application development cost, complexity, and time to market. It can also optimize operations across a multi-cloud environment for IT efficiency, security, and agility that supports business revenue growth.

VMware Aria Automation can help transform existing manual workstreams associated with the application life cycle, inconsistent policy, and tooling across different applications and environments. It can also help mitigate the risk of inconsistent security and compliance that could compromise the overall security of the data center.

This chapter helps you to take advantage of Aria Automation. You'll discover how to apply the five strategic automation use cases for your business and gain a competitive advantage.

Self-Service Multi-Cloud

If you work in IT or support, you'll be familiar with a common catch-22. Should you pay attention to immediate business needs, especially around the restrictions of your current environment, or focus on the latest developments in technology that can support productivity and value longer-term?

Adopting self-service automation can deliver numerous benefits and help IT transform for the future. And the good news is that you may already have most of what's required in place. Self-service is clearly not a new concept, so why pay attention to it now?

Clearly, the COVID-19 pandemic has accentuated an existing need to transform IT. Self-service automation has taken center stage because it is seen as both an efficiency and transformation enabler due to automation capabilities in areas such as resource provisioning, application deployment, and overall systems orchestration.

Customers can automate VMware Cloud Foundation to implement an on-premises, self-service private cloud or VMware Cloud (for example, VMware Cloud on AWS) to implement a self-service public cloud. Aria Automation offers a single VMware Cloud API for infrastructure as a service (IaaS) along with cloud organization constructs that uniquely delineate the IT operator, along with consumer roles and responsibilities for VMware infrastructure.

REMEMBER

As a result, Aria Automation can enable rapid implementation of multi-cloud environments with a unified self-service provisioning layer along with a seamless workload portability across private and public cloud infrastructure. Furthermore, it offers the ability to apply the same self-service catalog, content, and policies, providing centralized and streamlined operations with unified visibility and management across multi-cloud environments while providing developers with a single, consistent digital foundation to interact with programmatically.

In the context of self-service multi-clouds, Aria Automation provides:

- **Portability:** Leverage a unified self-service provisioning and catalog layer with seamless workload portability across private and public clouds.
- **Flexibility:** Enable automated infrastructure provisioning by providing a self-service catalog and pipelining infrastructure for cloud admins, and manage it with governance policies for better insights and control.
- **Choice:** Build designs from scratch using a rich set of building blocks (cloud-agnostic provisioning, AWS infrastructure-as-a-service, AWS services), or request designs and images from third-party platforms via the self-service catalog.

DevOps for Infrastructure

DevOps started as a buzzword that eventually managed to gain enough traction to become a movement. The idea of DevOps came from developers who decided to apply their powerful toolchain and agile processes to traditional IT Ops responsibilities.

When you hear about DevOps for Infrastructure through the lens of Aria Automation, it's really about how you can apply the principles and mechanics of DevOps to the delivery of an infrastructure automation platform that provides services to support the needs of internal development teams.

TECHNICAL STUFF

The DevOps cycle is a continuous loop, broken down into various stages. The stages have different definitions, but the majority converge on plan, code, build, test, release, deploy, operate, and monitor.

TIP

It's important to understand the key principles that most DevOps implementations incorporate: automation, iterative development, self-service, collaboration, knowledge sharing, observability, and traceability. Don't confuse the DevOps principles with the DevOps mechanics (cycles and phases).

In the world of infrastructure, this typically boils down to three things:

- **Infrastructure-as-Code (IaC):** The ability to express and manipulate infrastructure the same way as application code. The goal of IaC is to automate the provisioning and maintenance of the infrastructure to the highest degree possible. What is in fact coded is the Ops' intent via policies and variables and end-user needs via inputs and tags.
- **Iterative development with GitOps:** The collaborative updating of the infrastructure, by the developers and/or operators, as infrastructure needs evolve. Iterative development can use imperative or declarative language and ensures proper collaboration and accountability by leveraging Git-based version control systems (VCS).

- **Infrastructure pipelines:** Continuous delivery pipelines, specifically built to manage infrastructure in an automated way. CI/CD tools are the skeleton of any DevOps implementation. Infrastructure pipelines bring the best practices from the CI/CD world to traditional IT processes. With direct integrations to IaC tools and low-code pipeline, interfaces can greatly help IT departments adopt DevOps practices.

Aria Automation delivers benefits in three areas: scale, speed to market, and reliability. Customers achieve a positive ROI via efficiency, higher productivity, and faster time to value — delivering infrastructure with faster, smaller, and more frequent release cycles that allow scalability and reliability. In addition, I&O teams can satisfy the iterative development needs of developers, SREs, and DevOps engineers with frictionless governance for a better software quality and/or customer satisfaction.

REMEMBER

These drivers are equally important in the infrastructure as they are in the application world. Typically, infrastructure delivery needs to be in lockstep with software delivery. Otherwise, IT becomes the bottleneck. And, for better or worse, in today's cloud world, IT bottlenecks are quite easy to bypass.

Kubernetes Infrastructure Automation

Kubernetes is the most popular technology in IT today. It has been the rising star for the last couple of years and is gaining traction within the IT community as the new standard for deploying applications.

VMware has incorporated Kubernetes into its strategy. It is the technology that's evolving the application runtime space. Previously, the virtual machine was the preferred delivery method, but now the container is becoming the new normal. This doesn't mean virtual machines will go away. The result is that providers of infrastructure resources now need to be able to provide two things to their consumers: virtual machines and containers enabled by Kubernetes.

The goal is twofold:

- To productize and industrialize Kubernetes so that it's easy to consume, deploy, and manage by both developers and

infrastructure operators, with or without extensive Kubernetes experience

- To create a similar experience to the management and operations of virtual machines

At the management layer, all infrastructure and application resources need to be managed. The Aria and CloudHealth product portfolio provides capabilities to manage Kubernetes from a cost, security, and compliance perspective. Specifically, Aria Automation provides a platform that enables virtual administrators to manage Kubernetes the same way they've traditionally managed virtual machine workloads.

As a result, the customer can enable Kubernetes cluster management, self-service, and application deployment in a mixed, virtualized, and Kubernetes infrastructure through a central management plane with a unified operating/governance and consumption model. It can provide the ability to manage and govern Kubernetes clusters and namespaces, as well as discover and import clusters; empower developers to request Kubernetes clusters and namespace self-service from a catalog; and enable Kubernetes application deployment on clusters from pipelines.

Aria Automation provides numerous benefits for Kubernetes infrastructure automation, such as the ability to:

- **Streamline development and agile operations:** Accelerate innovation for modern cloud native applications.
- **Take advantage of existing investments in VMware technology and skillsets:** Gain flexibility with extensible integrations with VMware Tanzu Kubernetes Grid Integrated Edition, VMware vSphere with Tanzu, and Red Hat OpenShift Container Platform.
- **Simplify Kubernetes management for operators:** Ensure consistency in the application of your operational requirements.

VMware is investing heavily in its software portfolio to extend existing solutions and build new ones for Kubernetes. VMware can support businesses when it comes to building, running, and managing Kubernetes at scale within your IT organization, whether that's in the private, hybrid, or public cloud.

Network Automation

With users worldwide communicating, collaborating, and transacting through business-critical applications and services, organizations everywhere are pursuing new and differentiated business models and revenue sources that require a modern, agile IT infrastructure. Organizations need to release innovative and updated applications and services more frequently, and with enhanced reliability and security.

Network automation is the key to integrating multiple parts of the IT organization — from network engineering to cloud operations — and to modernizing IT processes across the IT delivery cycle to gain a competitive advantage.

REMEMBER

VMware has a unique and powerful network automation solution that enables faster deployment and completes life cycle automation of traditional and modern applications with networking and security services. By enabling consistency across clouds, VMware helps organizations achieve faster time to market, operational savings, productivity gains, and business resiliency.

VMware network automation automates VMware NSX with VMware Aria Automation. It combines the Aria Automation modern infrastructure automation platform with NSX network virtualization to enable rapid application rollout with networking and security services. By applying DevOps principles to network infrastructure delivery, this solution ensures that network policies are managed with workloads to eliminate operational bottlenecks in the application life cycle. This solution enables fast, consistent networking and better security for VM- and container-based workloads across private, hybrid, and multi-cloud environments.

Here are the benefits:

- **Gain business agility, supporting faster delivery of applications:** Support the business rollout of new products and services and move into new markets, while reducing CapEx and OpEx.
- **Establish consistent infrastructure and operations:** Enable a multi-cloud strategy and support both traditional and modern cloud native applications.

- **Quickly configure consistent networking and security across applications, environments, and clouds:** Reduce application provisioning time from weeks to minutes, while ensuring standardized environments and avoiding configuration drift.

With network automation, you can leverage faster deployment and complete life cycle automation of traditional virtual machine (VM) and modern container-based applications with consistent networking and security services across private, hybrid, and multi-cloud environments.

Security Operations

Security operations is a collaboration between IT security and operations teams that integrates tools, processes, and technology to keep an enterprise secure while reducing risk.

As information security teams become more important in organizations, a distinct gap often arises between them and IT operations teams. Each has fundamentally different priorities, which can result in conflicting efforts and disparate tools that create inefficiencies, reduce security postures, and open an organization to greater risks.

REMEMBER

Much like DevOps before it, SecOps (security plus operations) is a movement created to facilitate collaboration between security and operations teams and integrate the technology and processes they use to keep systems and data secure. Security and IT operations teams must work together to keep modern data centers compliant and secure, but their efforts are often crippled by disparate tool sets, misaligned workflows, and competing priorities. It's time for that to change. Aria Automation for Secure Clouds is a powerful add-on component for Aria Automation that gives IT operations and security teams the automation tools and content they need to build and maintain secure, compliant IT infrastructure on-premises or in the cloud.

Aria Automation for Secure Clouds provides continuous operating system compliance enforcement, an automated vulnerability detection and remediation feature, plus immediate insights into the state of your IT systems.

Aria Automation for Secure Clouds helps to:

- **Enforce security:** Remediate critical security threats across on-premises or cloud infrastructure with powerful vulnerability remediation automation.
- **Maintain compliance:** Use out-of-the-box Center for Internet Security (CIS) certified content to provision IT systems that start compliant and stay that way.
- **Reduce risk:** Employ powerful SecOps automation that goes beyond scanning to find and fix critical IT vulnerability and compliance issues.

With Aria Automation for Secure Clouds, security and operations teams can work together to define a corporate IT security policy, scan systems against it, detect vulnerabilities and non-compliance issues, and actively remediate them — all from a single platform.

REMEMBER

VMware Aria Automation is an infrastructure automation platform that:

- Enables private and multi-cloud environments on VMware Cloud infrastructure.
- Delivers network automation, security operations, self-service cloud, DevOps for infrastructure, and Kubernetes automation capabilities that help you adopt modern automation practices to increase business and IT agility, productivity, efficiency, security, and compliance.
- Automates manual tasks to save time and budget, freeing up IT resources (already stretched thin due to other critical business challenges) to tackle the more strategic projects that drive business value.
- Integrates, streamlines, and modernizes traditional, cloud native, and multi-cloud infrastructures to simplify IT and prepare for the future of your business.

IN THIS CHAPTER

- Checking online for a wide array of options
- Watching videos and webinars

Chapter 8
Ten Resources to Get Started with Aria Automation

This book presents an introduction to Aria Automation and why it's an important topic for your organization. If you want to take a deep dive into it, help is at hand. This chapter presents a list of ten resources to enhance your understanding of Aria Automation and help you get started.

Websites

You can find a plethora of websites with helpful information. We suggest you begin with the following to complement this handy guide:

- **VMware Aria Automation:** `www.vmware.com/products/aria-automation.html`
- **VMware Aria Automation Config:** `www.vmware.com/products/aria-automation/saltstack-config.html`

- **VMware Aria Automation Documentation:** https://docs.vmware.com/en/vRealize-Automation/index.html
- **Learn Aria Automation:** https://learnvrealizeautomation.github.io/
- **Aria Automation Reddit Channel:** www.reddit.com/r/vRealize_Automation/

Analyst Research

Experts in the field of automation are a great resource. Get an independent analyst's view on the state of automation via these resources:

- **IDC White Paper: Automation, Analytics, and Governance Power Enterprise Multicloud Management Strategies:** www.vmware.com/content/microsites/learn/en/350237_REG.html
- **IDC: Worldwide IT Automation and Configuration Management Software Market Shares:** www.vmware.com/learn/612434_REG.html
- **Transform Your IT with Self-Service Delivery:** www.vmware.com/learn/728573_REG.html?cid=7012H000001YqqeQAC&src=wb_5fce34a662be7
- **DevOps Trends in Enterprise IT – A Forrester Opportunity Snapshot:** www.vmware.com/learn/672595_REG.html
- **Selecting a Hybrid and Multicloud Management Solution:** www.vmware.com/learn/683326_REG.html

Blogs/Publications

Many automation experts blog about the lessons learned and share example workflows. Follow their conversations on blogs:

- **VMware Cloud Management blogs:** https://blogs.vmware.com/management/
- **VMware Cloud blogs:** https://blogs.vmware.com/cloud/
- **Getting Started with the Aria Automation Terraform Provider:** https://blogs.vmware.com/management/2020/01/getting-started-with-vra-terraform-provider.html
- **VMware Aria Automation:** https:/blogs.vmware.com/management/2021/08/whats-new-with-vrealize-automation-technical-overview.html
- **VMware Aria Automation Orchestrator and Action Based Extensibility:** www.vmware.com/products/aria-automation-orchestrator.html

Webinars

Webinars are another great resource to keep informed about automations. Check out the following:

- **Aria Automation – What's New:** www.brighttalk.com/webcast/18420/466845
- **Introducing Aria Automation Config:** https://bit.ly/3KdCsDk
- **Defining and Delivering DevSecOps Across Your IT Organization:** https://bit.ly/3tuKcuu
- **Introducing Terraform in Aria Automation:** https://bit.ly/3I1N2ep
- **Overcome Fear and Uncertainty To Benefit From DevOps:** https://bit.ly/3zWFbfu
- **VMware events and webinars:** www.vmware.com/company/events.html

Discussion Groups

You can join a discussion group to post questions and connect with your fellow automation engineers in the industry. Here are some discussion groups to try:

- **Red Hat Ansible Community:** `www.ansible.com/community`
- **Terraform Discussion:** `https://discuss.hashicorp.com/c/terraform-providers/vmware/39`
- **VMware{code}:** `https://code.vmware.com/`

Online Courses

You can also take an online class to enrich your understanding of automation. Here are some of the best, many of which are free or inexpensive:

- **Networking and Security Architecture with VMware NSX:** `www.coursera.org/learn/networking-security-architecture-vmware-nsx`
- **DevOps Foundations: Infrastructure As Code:** `www.linkedin.com/learning/devops-foundations-infrastructure-as-code`
- **Introduction to IT Automation:** `www.linkedin.com/learning/introduction-to-it-automation`
- **Automation with Azure Powershell and ARM Templates:** `www.linkedin.com/learning/automation-with-azure-powershell-and-arm-templates`
- **VMware Cloud: Deploying and Managing on AWS:** `www.linkedin.com/learning/vmware-cloud-deploying-and-managing-on-aws`

Podcast Feeds

Podcasts are another great way that you can learn from leading automation experts. Here are a few to start with:

- **VMware Aria Automation Ask an Expert podcast:** https://soundcloud.com/vmware/sets/vmware-vrealize-automation-ask
- **VMware Podcasts:** https://bit.ly/33a97sN

Videos

You can find a wealth of videos on network automation from practitioners and trainers. Check these out:

- **Aria Automation Media:** www.vmware.com/products/aria-automation/media.html
- **Aria This Live! (VMworld 2021 recap):** www.youtube.com/watch?v=PMHRYs482Jo
- **DevOps for Infrastructure (multiple videos):** www.vmware.com/products/aria-automation.html?resource=resource-listing%3Aoverview
- **Aria Automation overview (multiple videos):** www.vmware.com/products/aria-automation.html?resource=resource-listing%3Aoverview
- **Aria Automation Config (multiple videos):** www.vmware.com/products/aria-automation/saltstack-config.html

Books

When you're ready to take a deeper dive into automation, why not get the blueprint from the technical experts to help you understand what's going on under the hood? Here are some book recommendations:

- ***Network Automation For Dummies,* VMware Special Edition:** www.vmware.com/content/microsites/learn/en/544472_REG.html

- ***Cloud Automation For Dummies,*** **VMware Special Edition:** www.vmware.com/content/microsites/learn/en/40686_REG.html
- ***Intelligent Automation with VMware,*** **by Ajit Pratap Kundan:** www.google.com/books/edition/Intelligent_Automation_with_VMware/086PDwAAQBAJ?hl=en&gbpv=0
- ***Infrastructure As Code*****, by Kief Morris:** www.google.com/books/edition/Infrastructure_as_Code/Wz2KzQEACAAJ?hl=en
- ***Network Automation Made Easy*****, by Ivo Pinto:** www.google.com/books/edition/Network_Automation_Made_Easy/ocySzgEACAAJ?hl=en
- ***The Cloud Computing Book*****, by Douglas Comer:** www.google.com/books/edition/The_Cloud_Computing_Book/7Ag0EAAAQBAJ?hl=en&gbpv=0

Conferences and Meetups

The best way to learn a new technology is to get hands-on experience by going to a conference. Many conferences offer low-cost, pre-conference training workshops, including the following:

- **AnsibleFest:** www.ansible.com/ansiblefest
- **ChefConf:** www.chefconf.io/
- **HashiConf:** https://hashiconf.com
- **Puppetize Digital:** https://puppet.com/puppetize/
- **PyCon US:** https://us.pycon.org/
- **Red Hat Summit:** www.redhat.com/en/summit
- **SaltConf:** https://saltconf.com/
- **VMworld:** www.vmworld.com/en/us/index.html

Welcome to a bold new approach to technological practices

To embrace the digital transformation, businesses need technology that adapts and scales rapidly, reliably, and cost-effectively. Applications are more important than ever, progressively serving as the means by which organizations engage with customers. As a result, enterprises require more agility and speed throughout their digital processes, including the workflows involved in the designing, deployment, and management of applications.

Help is at hand with *Aria Automation For Dummies*. Discover how to pivot away from traditional workload management toward a more agile infrastructure and application configuration, with VMware's modern, event-driven infrastructure automation platform that can improve productivity and efficiency, and ultimately transform your business.

Inside...

- Understand the need for automation
- Increase IT agility and productivity
- Avoid downtime, data loss, and security breaches
- Ensure security and compliance
- Improve time to market and customer satisfaction
- Explore analyst research, blogs, courses, and podcasts

vmware®

Go to Dummies.com™
for videos, step-by-step photos, how-to articles, or to shop!

ISBN: 978-1-394-23466-0
Not For Resale

for dummies®
A Wiley Brand

WILEY END USER LICENSE AGREEMENT

Go to www.wiley.com/go/eula to access Wiley’s ebook EULA.