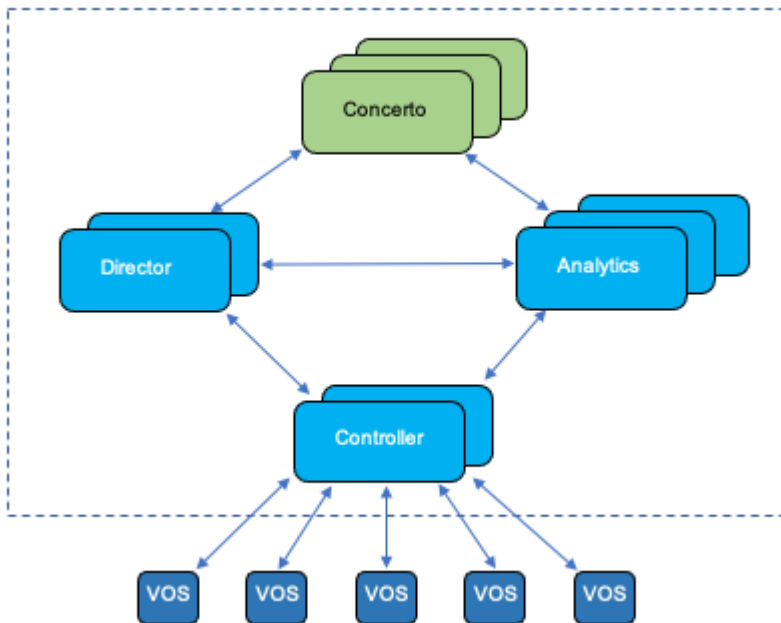

Versa Concerto Overview

 For supported software information, click [here](#).

The Versa Concerto™ orchestrator provides an easy-to-use user interface to configure and monitor Versa OS™ (VOS™) devices in Secure SD-WAN and secure access service edge (SASE) deployments. The Concerto orchestrator microservices architecture allows it to scale to manage tens of thousands of VOS devices. The Concerto orchestrator uses the services of Versa Director, Versa Controller, and Versa Analytics (collectively called the *DCA complex*) in managing VOS devices. It is an orchestration layer that sits above the current Versa DCA components, as shown in the following figure.



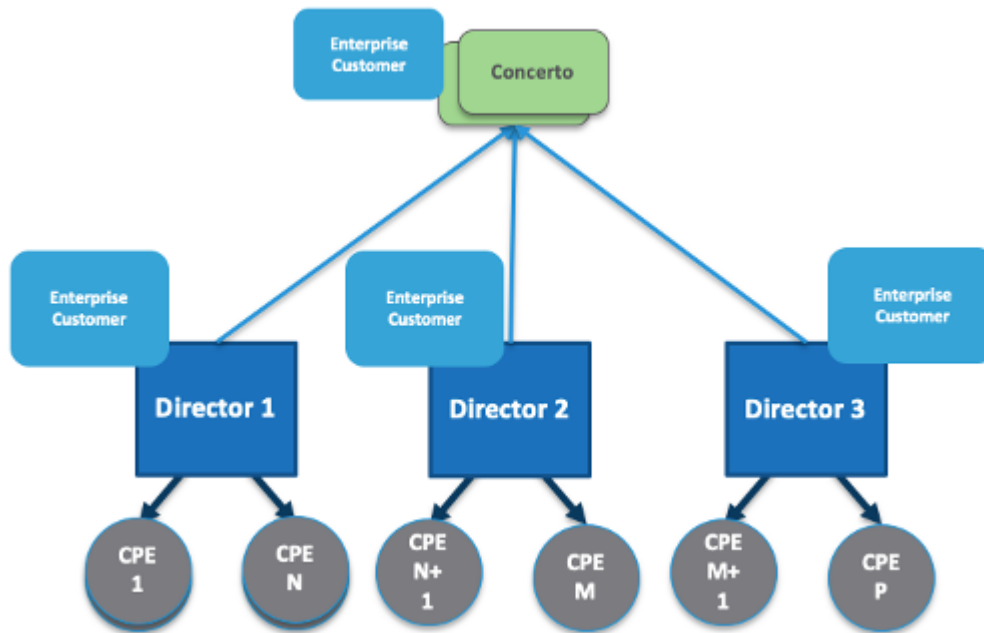
The following sections describe the high-level features of the Concerto orchestrator.

Single Pane of Glass

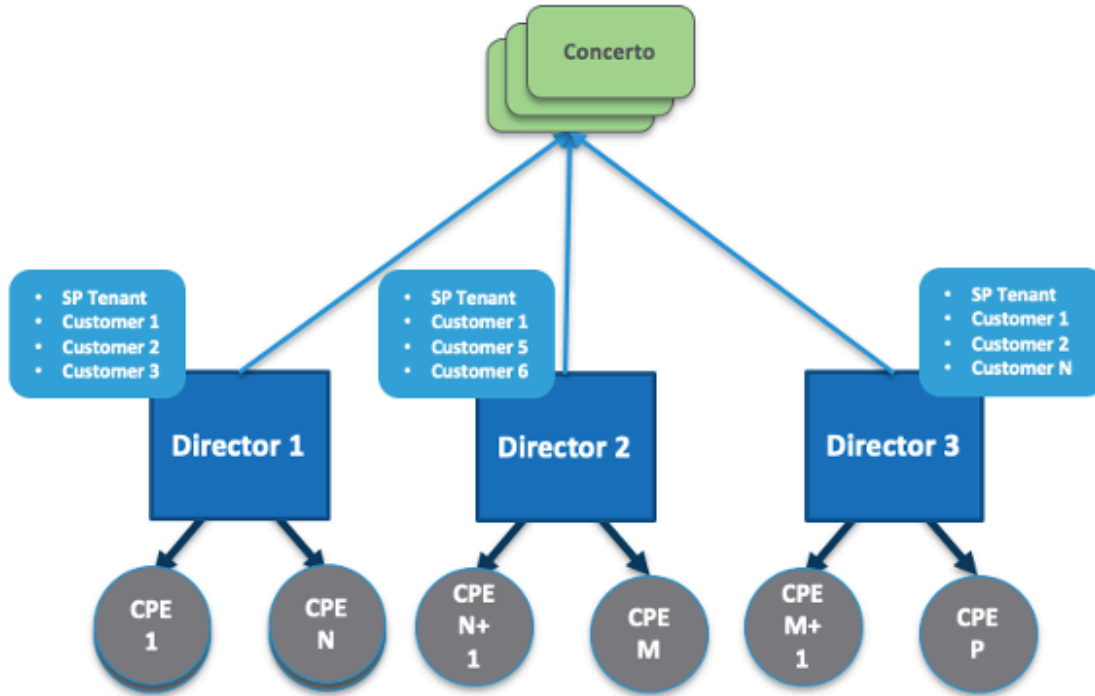
With its microservices architecture, the Concerto orchestrator can scale horizontally, connecting to multiple Versa DCA complexes. The Concerto orchestrator provides a single pane of glass for managing multiple tenants and VOS devices

spanning multiple DCAs.

In large-scale enterprises, the enterprise has more VOS devices than a single DCA can manage, so these devices are distributed across multiple DCAs that can be managed by a single Concerto orchestrator.



In service provider deployments with many customers, the service provider has a large number of VOS devices and multiple customers. The customer tenants and their VOS devices are distributed across multiple DCAs, and the service provider administrators choose the DCAs with which a given customer tenant is associated. Here, too, the multiple DCAs can be managed by a single Concerto orchestrator.



Lifecycles

The Concerto orchestrator is organized around *lifecycles*, which you use to perform the tasks required to manage DCA complexes. The following are the Concerto lifecycles:

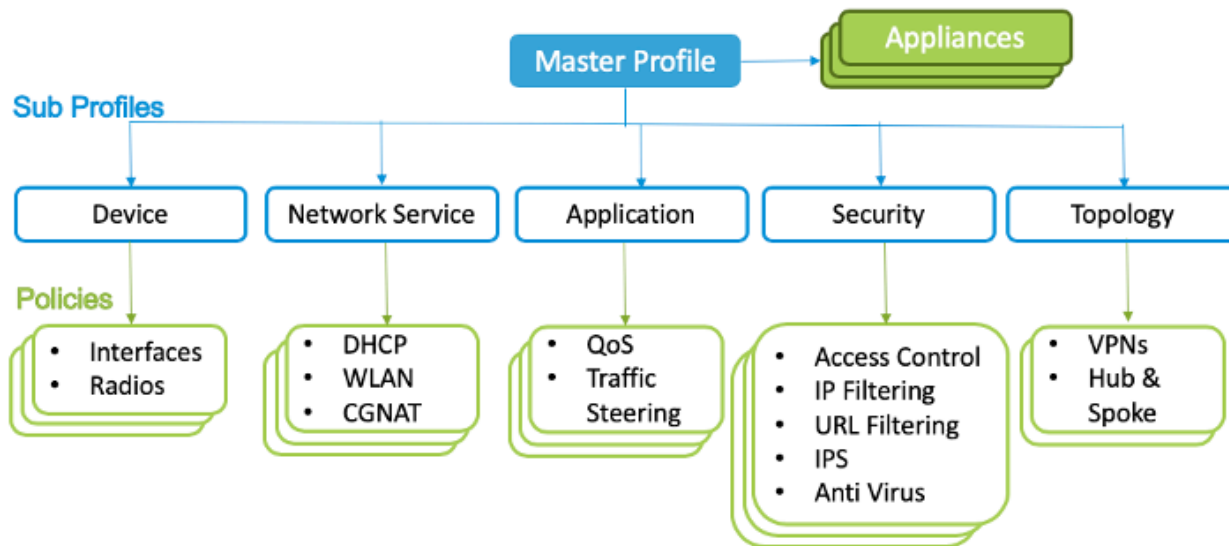
- **View**—(For Releases 11.3.1 and later.) Displays a unified view of the Versa Secure Private Access (VSPA), Versa SD-WAN, and Versa Security monitor screens.
- **Configure**—For creating all the configuration objects, including master profiles, subprofiles, policies, and rules.
- **Deploy**—For creating sites and devices within sites, associating devices with master profiles, and configuring bind variables. The Deploy lifecycle also publishes the configuration to the Director node and the VOS device (if ZTP has completed on the device).
- **Monitor**—Displays real-time status information for the enterprise network, individual site, or selected VOS device. The Monitor lifecycle has various views for topologies, applications, and users.
- **Analytics**—Displays historical and detailed information. The Analytics lifecycle shows the entire Analytics portal.
- **Inventory**—Displays information about all the devices in a searchable table. The information includes location, serial number, model number, software version, add-on hardware components, and firmware version.
- **Users**—For managing predefined and custom roles and associating users with roles.
- **Settings**
 - For service provider administrators, use to set a global SASE domain.
 - For enterprise administrators, use to set an IPsec branch-to-branch transform and an IPsec branch-to-branch Diffie-Hellman group for a tenant.

All seven lifecycles are available for each tenant. They lifecycles are displayed in Tenant view in the left navigation pane on the Concerto portal, as shown in the following figure.



Configuration Hierarchies

In the Concerto portal, a device's configuration is maintained in a master profile. The master profile is organized into configuration subsections that consist of subprofiles, policies, and rules. The following figure illustrates the master profile organization.



A master profile is a collection of one or more subprofiles. A single master profile can be applied to one or more devices.

The master profile can contain five types of subprofiles:

- Device
- Network service
- Application
- Security
- Topology

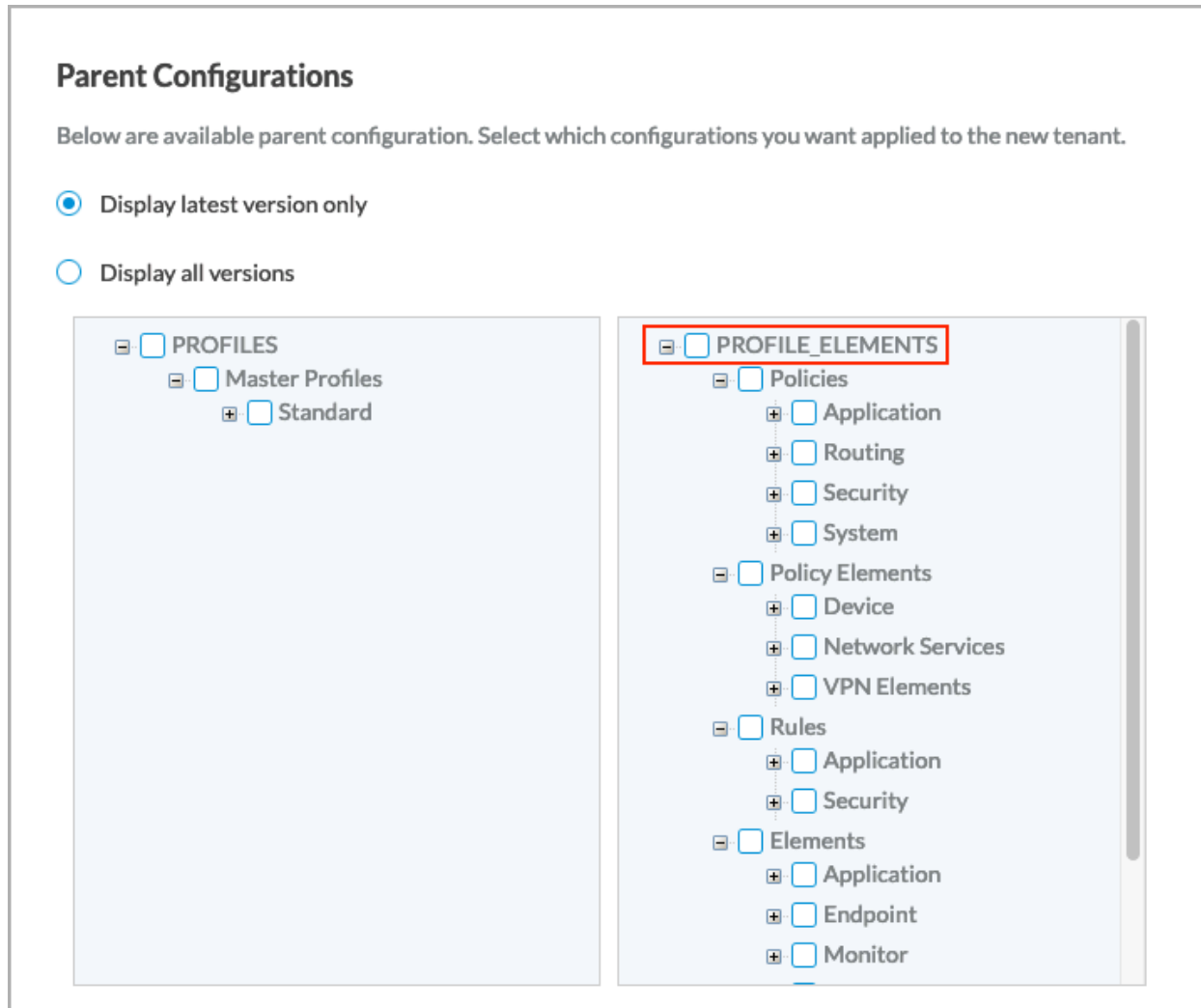
Each subprofile is a collection of different types of policies related to the subprofile. For example, a device subprofile is a collection of one or more interface and radio policies.

Each policy is a collection of rules or profile elements. For example, an interface policy in a device subprofile is a collection of WAN and LAN interfaces (profile elements), and an access control policy in a security subprofile is a collection of one or more access-control rules.

You can create each configuration object type as a reusable object. For example, you can use a network service subprofile in more than one master profile.

Reusable Configuration Objects

You can reuse any configuration object—a master profile, a standard profile, and a profile element—and you can refer to any configuration object in more than one higher-level object. The following figure illustrates the reusable folder structures in the Configure lifecycle.



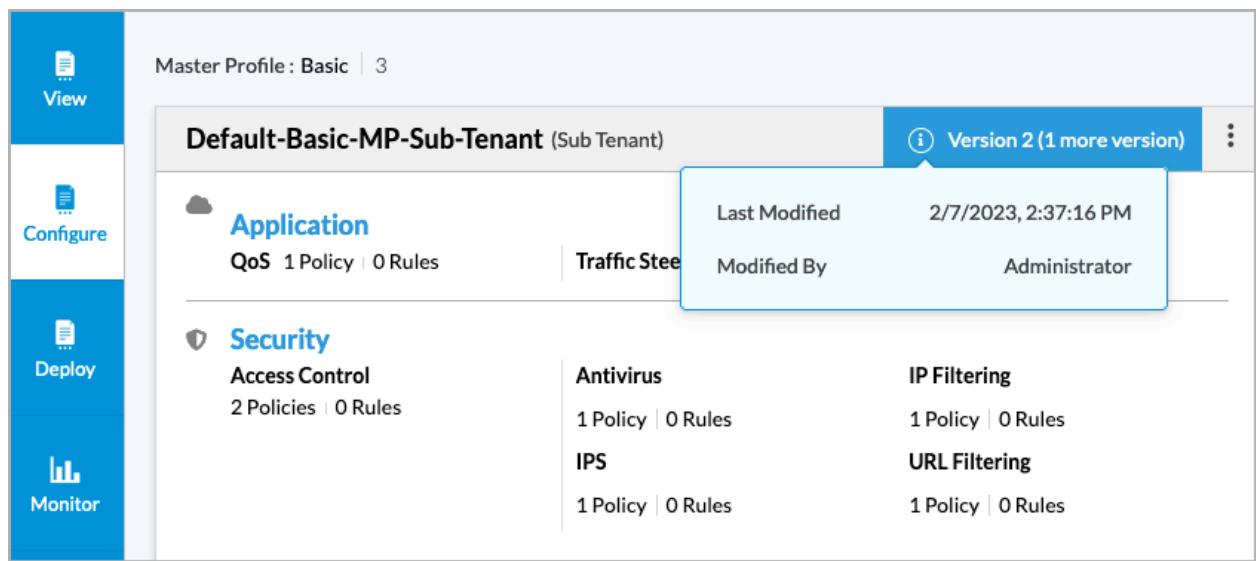
For Releases 10.2.1 and later, you can view all references to any configuration object (master profile, subprofile, or any profile element) from other levels of the configuration hierarchy. When you change the configuration of an object, you can then choose which versions of an object should be updated with the newer version. For more information about viewing references to objects, see [View References to Objects in the Configuration Hierarchies](#). For more information about propagating changes to an object, see [Propagate Object Configuration Changes](#).

Object Versioning

Each configuration object has a version associated with it. When you make changes to the object, a new instance of the

object is created, a new version number is given to it and the original version remains unchanged. If an object is referenced from within another higher-level object, the reference is based on the both the object name and version. You must explicitly change the reference to the new version of a referenced object. In some hierarchies, when the object version is changed, the Concerto orchestrator shows references to older versions of the object so that you can change some of the references to the newer version if desired.

As shown in the figure below, there are two versions of the Default-Basic-MP-Sub-Tenant master profile in the Master Profile > Standard folder. The blue rectangle shows how many versions there are of the profile. You can click the blue rectangle to view the two versions of the master profile. The information button shows when the version was last modified. You can apply different versions of the same profile to different devices.



For Releases 10.2.1 and later, the way in which configuration objects are versioned has been changed to reduce the number of versions that are created, which simplifies object configuration by creating fewer versions to track, improves performance, and saves storage space. For example, in previous releases, creating a new master profile could create up to ten or more versions before the master profile is finalized.

For Releases 10.2.1 and later, if you change a profile or profile-element configuration object, a new version of the object is not immediately created. To create a new version, you must attach the changed object to another object, such as a profile, subprofile, policy, or rule. For example, suppose you first create a new access control rule called AC-rule1. The initial version is called AC-rule1.v1. If you immediately make a change to this rule and save it, the version number stays the same because the rule has not been used in another object. If you then create a new policy and use AC-rule1.v1 in the policy, the version is incremented and is saved as AC-rule1.v2.

For Releases 10.2.1 and later, object versioning is unique across a tenant. If a tenant changes an object, the new version of the object is associated with the tenant that created it. For example, if Tenant-1 changes an object called object.v1 that has already been attached to another object, the version changes to object.v2. If Tenant-2 changes the same object.v1, the new version is saved as object.v3. If Tenant-1 later changes object.v2, it is saved as object.v4. In this case, object.v1, object.v2, and object.v4 are unique to Tenant-1, while object.v3 is unique to Tenant-2. If you change

an element that is used in other objects, the system displays a prompt asking if you want to propagate this change to other objects that use the changed object. For more information, see [Propagate Configuration Changes to Appliances](#).

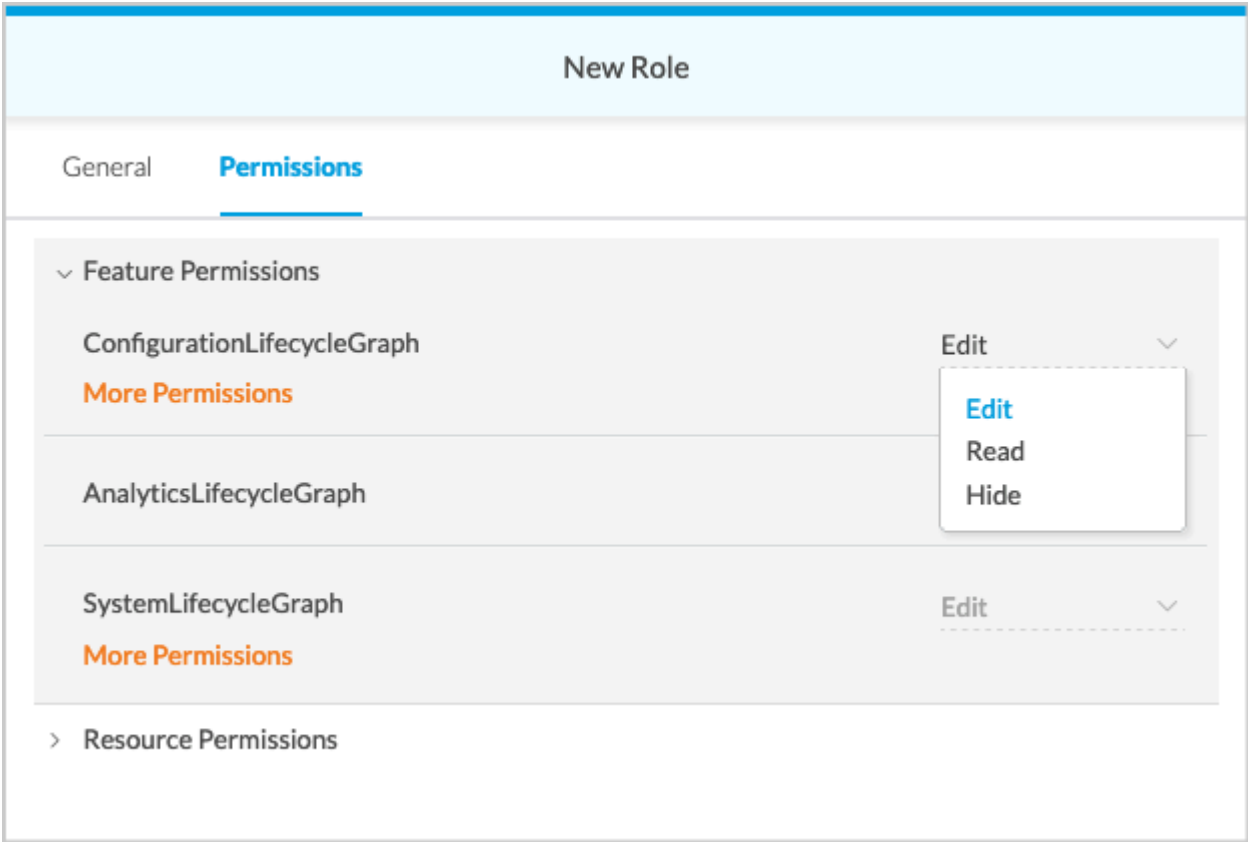
For Releases 11.3.2 and later, you can automatically delete SD-WAN object versions that are no longer used by an appliance and are not the latest version. For more information, see [Configure Automatic Deletion of SD-WAN Object Versions](#).

Custom Rules and Granular Permissions

The Concerto orchestrator has two provider-level default user roles—Service Provider Administrator and Service Provider Operator—and two tenant-level default user roles —Enterprise Administrator and Enterprise Operator. Administrators can also create custom user roles within these four roles and modify the permissions of the default roles.

The Concerto orchestrator implements permissions at the feature level (for example, all interface configurations within a tenant) and at the resource level (for example, a specific subset of interfaces). In addition, resource-level permissions allow different sites within a tenant to be managed by different administrators, and they allow different provider administrators to manage different tenants.

The following figure shows an example of the feature-level permissions that can be set when you create a new user role.



The following figure shows an example of resource-level permissions. For the role of a tenant Enterprise Administrator,

some sites are marked with Edit permission, some are marked with Hide permission, and some are marked with Read permission.

Permissions		Enterprise...ator.v1
<input type="checkbox"/> Select All Sites 8		
> New York (Reset)	Edit	▼
> Miami (Reset)	Read	▼
> TEST-SITE-Y (Reset)	Edit	▼
> Test2-Site (Reset)	Hide	▼
> San Francisco (Reset)	Edit	▼
> Seattle (Reset)	Read	▼
> Test-Site (Reset)	Hide	▼
> Dallas (Reset)	Edit	▼

Each configuration object is a resource, and you can set permissions on an individual object. By default, permissions are automatically inherited from the parent object so that you do not need to set permissions for every object. You can, however, change the permissions for any resource. Also, you can assign multiple roles to a single user, with each role having its own set of permissions, thereby avoiding the need to create numerous custom roles.

In Releases 11.4.1 and later, a subtenant can inherit a role created in the parent tenant, but the role cannot be edited or deleted from the subtenant. It can be modified and deleted only from the parent tenant. If the parent tenant role is attached to a user in a subtenant, the role cannot be deleted from the parent tenant unless the same role is disassociated from the user in the subtenant.

Importing Configuration Objects into Child Tenants

For service provider deployments, the Concerto orchestrator allows you to build and test configuration objects once for each supported deployment and to import the configuration objects into customer tenants. You can also do this for enterprise deployments in which the enterprise has multiple tenants. The following figure shows the parent tenant resources when a subtenant is created. You can select some of the resources from the parent tenant and import them into the subtenant.

Parent Configurations
Below are available parent configuration. Select which configurations you want applied to the new tenant.
☒ Display latest version only ☐ Display all versions

- ☐ PROFILE_ELEMENTS
 - ☐ Elements
 - ☐ Application
 - ☐ Application Classification
 - ☐ Business-Critical-v1
 - ☐ Default-v1
 - ☐ Low-Priority-v1
 - ☐ Real-Time-v1
 - ☐ Application Group
 - ☐ Custom Application
 - ☐ Endpoint
 - ☐ Address Group
 - ☐ customservice-v1
 - ☐ Finance-v1
 - ☐ PhonesGroup-v1
 - ☐ tag99-v1
 - ☐ tag123-v1

Parameterized Variables with Types

You can parameterize any attribute in a configuration object. To parameterize a value, you enter the variable name starting with \$ in the UI text field. Each variable is assigned a type automatically based on the attribute. For example, for an IP address in a field, the variable type is IP address.

When two different attributes of different types have the same variable name, the Concerto orchestrator creates two different variables that have the same name but different types.

The following figure below shows the parameterization of a LAN interface IP address to \$Enterprise-LAN-Address.

Enterprise-LAN.v1

General

Address and Routing

QoS

Permissions

Address

IPv4 Address

\$Enterprise-LAN-Address

STATIC

DHCP

DHCP Relay

VRRP

Add IPv6 Address

Guest Interface

VPN Name

VERSA-Enterprise

> Static Routes

> Routing Protocol

Close

Next

The following figure shows variable names by type that are used in the master profile Asia-MasterProfile. The red dots preceding the variable name indicate that a value is missing and that you must provide a value.

Asia-MasterProfile

Version 2

Profile Name	Type	#Policies
Default-Device-SPv2	Device	2
Default-Mesh-Profile.v2	Topology	1
Default-Netwo...vice-SPv2	Network Service	2
Asia-Security-SPv2	Security	3

Variables | 6

Interface IP | 3

IP Address | 2

\$Interface IP 3

Name	Value
Corp-WiFi-Address	-
MPLS-Intf-Address	-
+1 More	

For Releases 10.2.1 and later, you can parameterize an interface's Location field in order to map ports to different VNI numbers on a port-by-port basis. Instead of using the pull-down menu to select a VNI interface, you can enter a variable string, for example, \$vniInterface, as shown below.

Edit Interface

Internet-1.v1

General

Connection

QoS

Permissions

Name

Internet-1

Version 1

Description

Type

Physical

Virtual

Enabled

Block ICMP

Category

WAN

Sub Category

Wired

Location

\$vniInterface

VLAN ID

\$Internet-1-VLAN

Link

Speed

Mode

MTU

AUTO

AUTO_DUPLEX

68-9000

Tags

Cancel

Next

In the Variables section of the profile screen, you then use the pull-down menu to select a VNI interface with which to populate the variable.

Deployment Modes

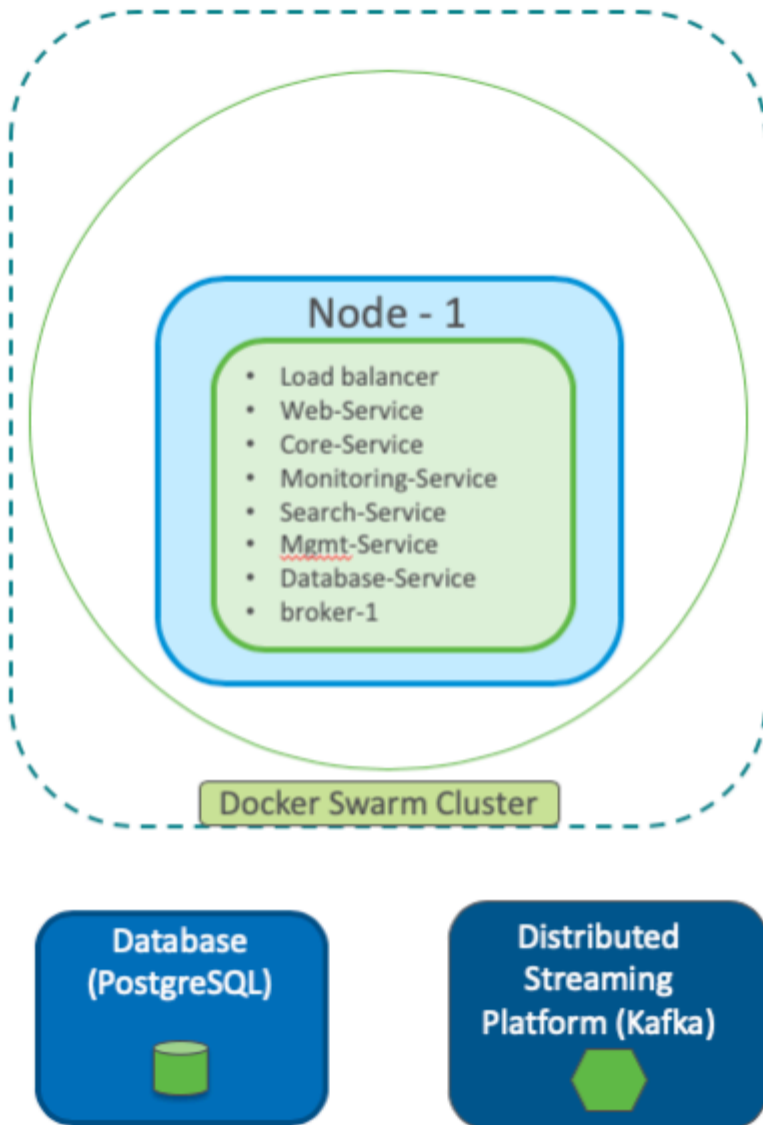
You can deploy the Concerto orchestrator as a single node or in multiple modes. This section describes some of the Concerto deployment modes.

Single-Node Lab Deployment

In lab deployments, which are typically created for functional testing and demos, you can deploy the complete set of Concerto containers in a single virtual machine (VM) or bare-metal server that has the resources listed below.

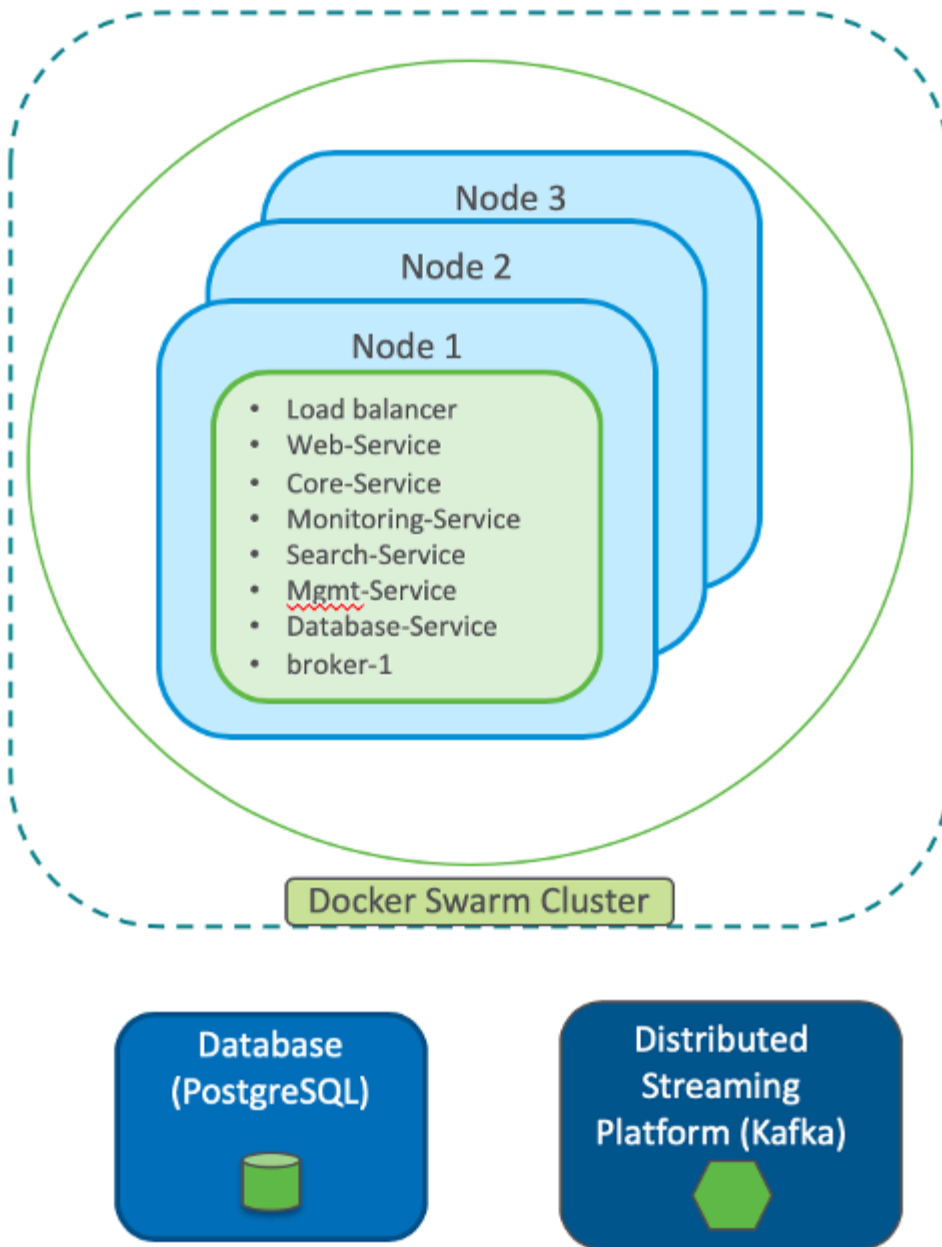
Container	Requirement
CPU	Minimum of 8 cores, with no hyperthreading enabled on the host
Memory	At least 16 GB of RAM
Disk	At least 150 GB of disk space
NIC	Two physical or virtual NICs, one for the northbound interface and the second for the southbound interface towards the DCA

The following figure shows the container services that run on a single node.



Multiple Nodes without Geographic Redundancy

You can use a multiple node deployment without geographic redundancy for both lab and production environments that have local redundancy and horizontal scaling. Within the same data center, you should deploy at least three Concerto nodes. You can add more nodes to increase the Concerto capacity.



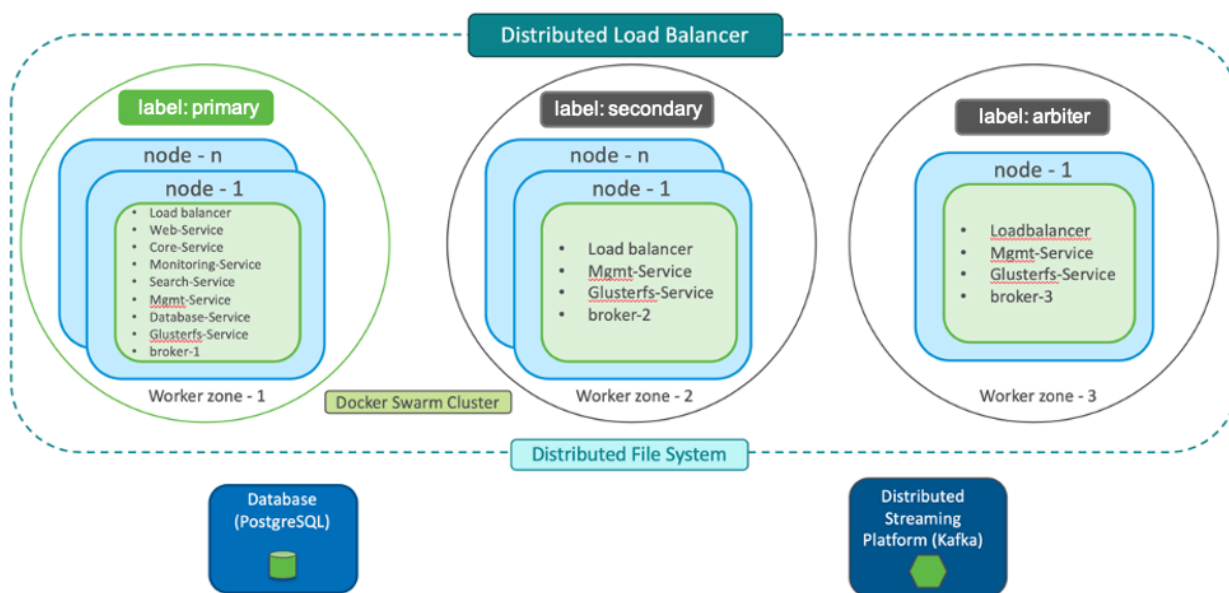
The required size of each node is shown in the following table.

Component	Requirement
CPU	Minimum of 8 cores without hyperthreading enabled on the host
Memory	At least 16 GB of RAM

Component	Requirement
Disk	At least 250 GB of disk space
NIC	Two physical or virtual NICs, one for the northbound interface and the second for the southbound interface towards the DCA

Multiple Nodes with Geographic Redundancy

Multiple node with geographic redundancy is an extension of the preceding mode that provides geographic redundancy. The following figure illustrates this mode. Here, Zone 1 (the active zone) and Zone 2 (the standby zone) are deployed with an equal number of nodes to allow the Concerto orchestrator to run at full capacity. When all the nodes in Zone 1 fail, Zone 2 takes over as the active role. If the cluster does not need to run at full capacity in Zone 2, you can deploy fewer nodes in Zone 2. In this mode you should deploy an arbiter node, which can be a small VM with 4 cores and 4 GB of RAM in a third-party data center or in the cloud, to avoid a split-brain scenario between Zone 1 and Zone 2.



Supported Software Information

Releases 10.1.1 and later support all content described in this article, except:

- Release 10.2.1 adds support for updated object versioning functionality in which a change to an object's configuration does not immediately create a new version of the object, and parameterizing an interface's Location field to map ports to different VNI numbers on a port-by-port basis.
- Release 11.1.1 adds support for SASE deployments; configuring management servers; and, configuring the root domain for a SASE deployment.

https://docs.versa-networks.com/Management_and_Orchestration/Versa_Concerto_Orchestrator/01_Getting_Started/01_Ver...

Updated: Wed, 23 Oct 2024 08:55:15 GMT

Copyright © 2024, Versa Networks, Inc.

- Release 11.3.1 adds the View lifecycle.
- Release 11.3.2 adds support for the automatic deletion of SD-WAN object versions that are no longer used by an appliance and are not the latest version.
- In Release 11.4.1, a subtenant can inherit a role created in the parent tenant, but the role cannot be edited or deleted from the subtenant. If the parent tenant role is attached to a user in a subtenant, the role cannot be deleted from the parent tenant unless the same role is disassociated from the user in the subtenant.

Additional Information

[Concerto Home Screen Overview](#)

[Configure Automatic Deletion of SD-WAN Object Versions](#)

[Propagate Configuration Changes to Appliances](#)

[Propagate Object Configuration Changes](#)

[View References to Objects in the Configuration Hierarchies](#)