
Concerto Administration

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

This article describes how to create and manage backup files. It also describes the Concerto CLI commands and their options.

Create and Manage Backup Files

You use the **vsh** client interface to back up and restore data from the Concerto Postgre database. You can back up the Postgre database in two ways:

- Periodic backup—The Concerto Postgre database on every active node is periodically backed up to the default directory, `/var/versa/ecp/archives` every day at 12 a.m. (system time). The backup files that are created by this periodic background process have the tag “_Nightly” appended to their filenames and can be viewed using the **list** option. By default, Concerto maintains a rolling period of 10 days worth of backups; you can configure this rolling period by using the **limit** option.
- Manual backup—All users with sudo access can create manual backups and can to store them in their desired locations with user-defined filenames. These backup files are not subject to being deleted after 10 days as long as they are not stored in the default directory, `/var/versa/ecp/archives`.

You can create and manage Postgre backup files using the **vsh** client interface to do the following:

- View all back up and restore command options
- Create backup files on demand
- List backup files
- Limit backup files
- Configure the back up time period
- Restore a backup file

View All Backup and Restore Command Options

To view all options for the **vsh** database backup command, issue the following command:

```
admin@concerto-node1:~$ vsh database backup --help
Usage: vsh database backup [OPTIONS] COMMAND [ARGS]...
Options:
  --help  Show this message and exit.
```

Commands:
create folder_name : The full path of the backup directory
limit
list
restore name : backup file name with full file path location

Create Backup Files on Demand

To create an on-demand backup file at the default destination, issue the following command:

```
admin@concerto-node1:~$ vsh database backup create /var/versa/ecp/archives
```

To create an on-demand backup file on the home directory, issue the following command:

```
admin@concerto-node1:~$ vsh database backup create /home/admin
```

On-demand backup files have a filename ending tag, such as “_Manual.pgsql”. These files are saved to the local file system of the Concerto server at a location that you specify.

List Backup Files

Concerto creates periodic backup files automatically. To view these backup files, issue the following command:

```
admin@concerto-node1:~$ vsh database backup list

SIZE  FILENAME
280K  dbexport_2021-04-16_00-00-11_concerto-20210415-095209-d90a480-10.2.1_Nightly.pgsql
```

A backup filename has the following format:

```
dbexport_YYYY-MM-DD_HH-MM-SS_build-name_build-type.pgsql
```

- Numeric date and time of the build, in the format `YYYY-MM-DD_HH-MM-SS`
- Full build name with the version number
- Build type—Manual or nightly for periodic backups. The names of the files created by periodic backups are appended with a tag, such as “_Nightly.pgsql”.

Backup files are saved to the local file system of the Concerto server in the local directory `/var/versa/ecp/archives`.

Configure the Backup Time Period

To view the backup time period limit, in days, issue the following command:

```
admin@concerto-node1:~$ vsh database backup limit

Current database backups are stored up to 10 rolling period of days
```

This limit applies only to backup files in the default backup directory, `/var/versa/ecp/archives`. Any files older than this

time limit are deleted.

To change the backup time period limit, issue the following command:

```
admin@concerto-node1:$ vsh database backup limit days
```

For example:

```
admin@concerto-node1:$ vsh database backup limit 60  
Database backups will be stored up to 60 days
```

Restore a Backup File

To restore the data in a backup file, issue the following command, where *filename* is the name of the backup file to restore:

```
admin@concerto-node1:$ vsh database backup restore filename
```

To restore the files from a periodic backup, use the backup filename as listed in the **vsh database backup list** command. You do not need to specify the directory path.

To restore a custom manual backup, provide a full path to the .pgsql backup data file in the **vsh database backup restore** command.

Note: Restoring backup data from a build version different from the currently running build version might result in unexpected behavior, especially if there are any database-related changes between the two builds.

Connect and Reset the Database

Concerto provides two additional options to access and clean the database manually.

- To connect to the active Postgre database directly from any active node, regardless of the deployment model, issue the following command. You may have to provide the database password, not the Concerto login password.

```
admin@concerto-node1:$ vsh database connect database-name
```

- To reset the entire portal database and create an empty portal database, issue the following command.

Caution: Use this option only when absolutely necessary. While this option resets the entire portal database and creates an empty portal database, it does not delete the application configuration files or alter deployment settings.

```
admin@concerto-node1:$ vsh database reset
```

Integrate Prometheus with Concerto

You can export statistics to a Prometheus server and monitor them using a web browser, as described in the following sections.

Configure the Concerto Host

To configure the Concerto host:

1. SSH into the Concerto host machine, and then edit the `glances.conf` file.

```
| sudo vi /etc/glances/glances.conf
```

2. Add the following lines to the `glances.conf` file. Replace the host with the IP address of the host machine. You can use any available custom port on the host.

```
| [prometheus]
| host=localhost
| port=9091
| prefix=glances
| labels=src:glances
```

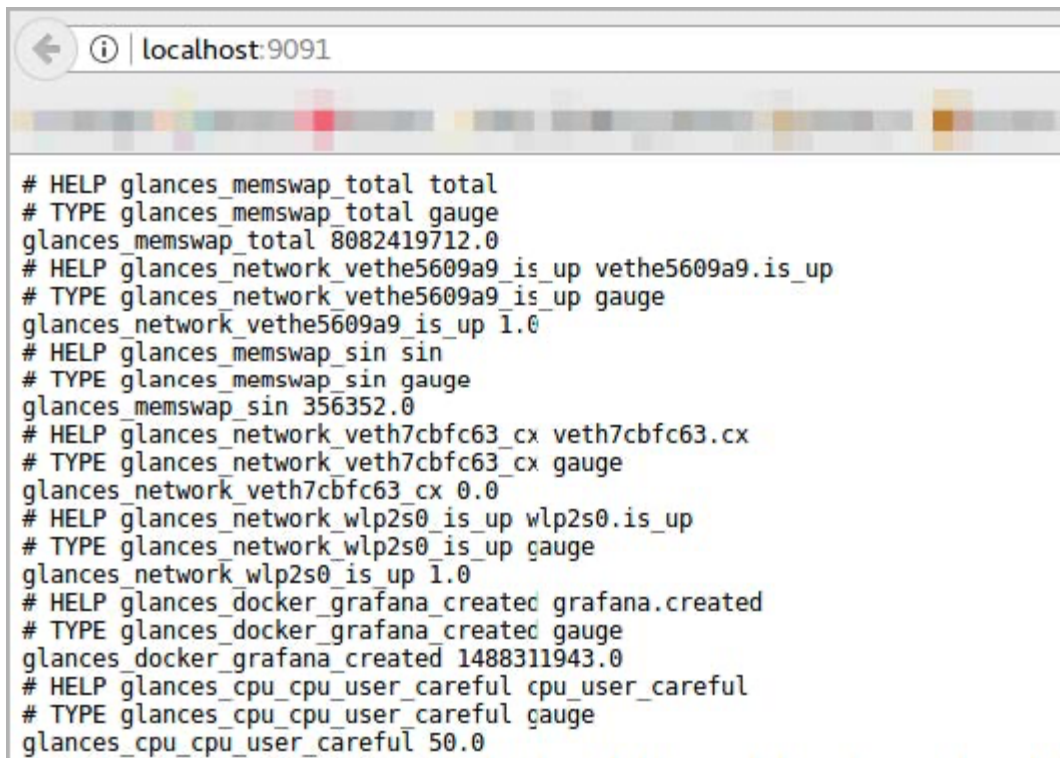
3. Stop and disable Glances.

```
| sudo systemctl stop glances.service
| sudo systemctl disable glances.service
```

4. Start Glances.

```
| sudo glances --export prometheus -q &
```

You can verify that Glances is exporting statistics by going to the URL <http://localhost:9091>. For example:



```
# HELP glances_memswap_total total
# TYPE glances_memswap_total gauge
glances_memswap_total 8082419712.0
# HELP glances_network_vethe5609a9_is_up vethe5609a9.is_up
# TYPE glances_network_vethe5609a9_is_up gauge
glances_network_vethe5609a9_is_up 1.0
# HELP glances_memswap_sin sin
# TYPE glances_memswap_sin gauge
glances_memswap_sin 356352.0
# HELP glances_network_veth7cbfc63_cx veth7cbfc63.cx
# TYPE glances_network_veth7cbfc63_cx gauge
glances_network_veth7cbfc63_cx 0.0
# HELP glances_network_wlp2s0_is_up wlp2s0.is_up
# TYPE glances_network_wlp2s0_is_up gauge
glances_network_wlp2s0_is_up 1.0
# HELP glances_docker_grafana_created grafana.created
# TYPE glances_docker_grafana_created gauge
glances_docker_grafana_created 1488311943.0
# HELP glances_cpu_cpu_user_careful cpu_user_careful
# TYPE glances_cpu_cpu_user_careful gauge
glances_cpu_cpu_user_careful 50.0
```

To stop exporting the statistics, issue the following command:

```
ps -aef|grep "sudo glances --export prometheus" | grep -v grep | awk '{ print $2 }' | xargs sudo kill -9
```

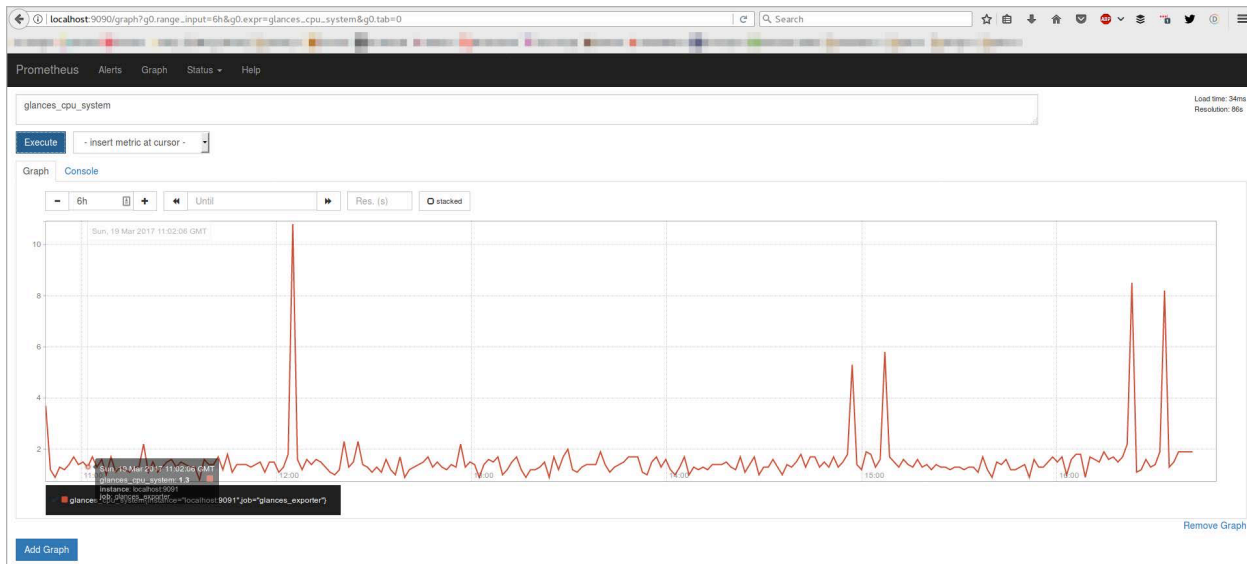
Configure Prometheus

To store the metrics in a Prometheus server, add an exporter to the Prometheus server configuration file, `prometheus.yml`, using the following lines:

```
scrape_configs:
- job_name: 'glances_exporter'
  scrape_interval: 5s
  static_configs:
  - targets: ['localhost:9091']
```

Replace the `localhost` and port for the Concerto host IP and port as specified in the `glances.conf` file in the [Configure the Concerto Host](#) section.

You can check the Prometheus at the URL `http://localhost:9090`.



Instructions Source: <https://glances.readthedocs.io/en/latest/prometheus.html>.

Grafana Support

Grafana is supported through Prometheus due to the fact that there is no timeseries database, such as InfluxDB, available on Concerto hosts.

Start a Local Prometheus Server

To run your own Prometheus server, use the following docker command after creating the prometheus.yml file with the exporter "scrape_configs" section, as shown above.

1. Create the Prometheus configuration file:

```
vi /home/admin/prometheus.yml
```

2. Add the following line for the DNS server to the /etc/resolv.conf file to download and install Prometheus container"

```
nameserver 8.8.8.8
```

3. Add the scrape configuration to the file and save it.
4. Change permissions for Prometheus configuration with scrape configuration:

```
chmod 774 /home/admin/prometheus.yml
```

5. Issue the following command:

```
docker pull prom/prometheus
```

6. Start the Prometheus docker container:

```
docker run -d --network host -v /home/admin/prometheus.yml:/etc/prometheus/prometheus.yml prom/
```

https://docs.versa-networks.com/Management_and_Orchestration/Versa_Concerto_Orchestrator/02_Common_Configuration/...

Updated: Wed, 23 Oct 2024 08:53:45 GMT

Copyright © 2024, Versa Networks, Inc.

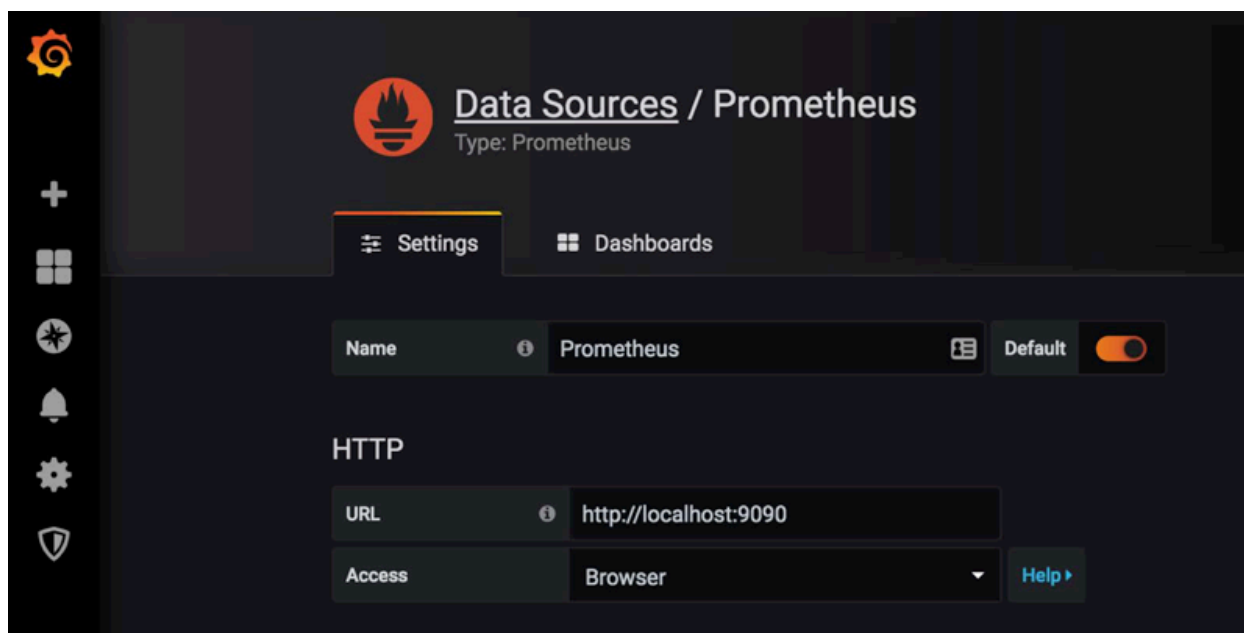
Follow Prometheus configuration instructions as specified above.

Create a Prometheus Data Source in Grafana

To create a Prometheus data source in Grafana:

1. Click on the "cogwheel" in the sidebar to open the Configuration menu.
2. Click Data Sources.
3. Click Add data source.
4. Select Prometheus as the type.
5. Set the appropriate Prometheus server URL (for example, `http://localhost:9090/`)
6. Adjust other data source settings as desired (for example, choosing the right access method).
7. Click Save & Test to save the new data source.

The following screen shows an example data source configuration:



Create a Prometheus Graph

Follow the standard way of adding a new Grafana graph, then do the following:

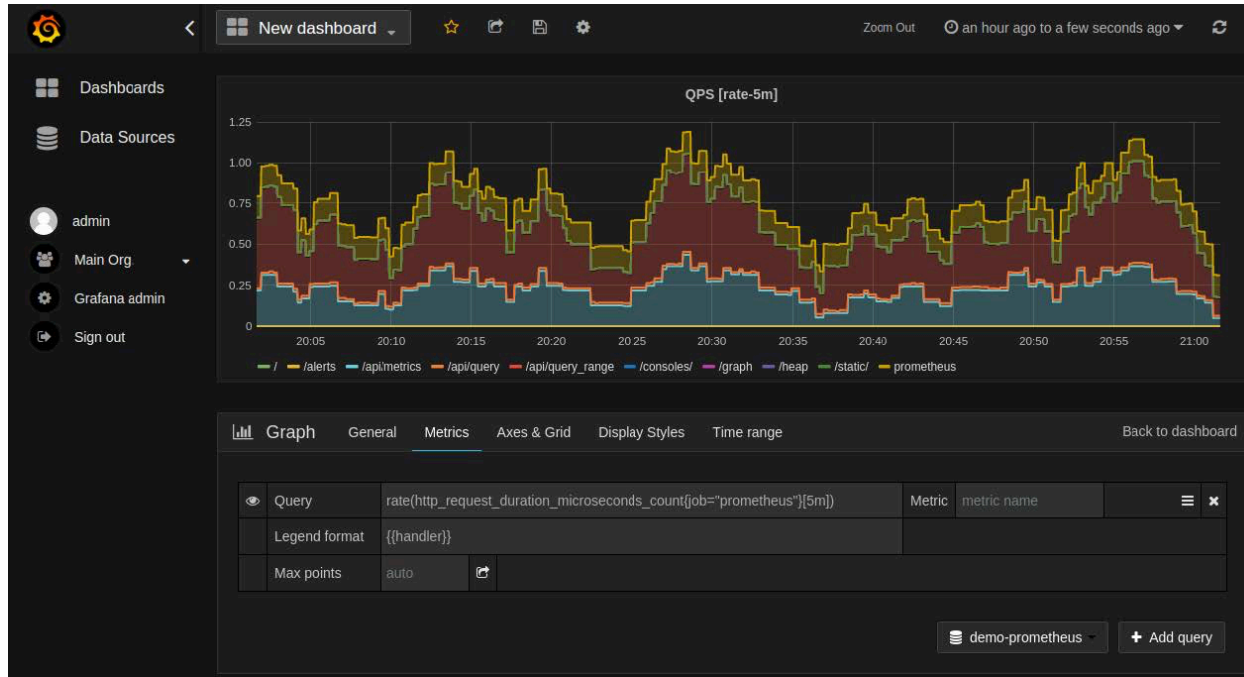
1. Click the graph title, then click Edit.
2. Under the Metrics tab, select your Prometheus data source (bottom right).
3. Enter any Prometheus expression into the Query field, while using the Metric field to lookup metrics via autocompletion.

- To format the legend names of time series, use the Legend format input. For example, to show only the method and status labels of a returned query result, separated by a dash, you would use the following legend format string:

```
{{method}} - {{status}}
```

- Tune other graph settings until you have a working graph.

The following shows an example Prometheus graph configuration.



In Grafana 7.2 and later, the `$__rate_interval` variable is recommended for use in the **rate** and **increase** functions.

Import Pre-built Dashboards from Grafana.com

Grafana.com maintains a collection of shared dashboards which can be downloaded and used with standalone instances of Grafana. Use the Grafana.com Filter option to browse dashboards for the Prometheus data source only.

You must currently edit the downloaded JSON files manually and correct the **datasource:** entries to reflect the Grafana data source name which you chose for your Prometheus server. Use the Dashboards → Home → Import option to import the edited dashboard file into your Grafana installation.

Instructions Source: <https://prometheus.io/docs/visualization/grafana/>

Concerto CLI Commands

The following CLI commands are available in Concerto 10.2.1.

- vsh system**—Display system commands and their options.

https://docs.versa-networks.com/Management_and_Orchestration/Versa_Concerto_Orchestrator/02_Common_Configuration/...

Updated: Wed, 23 Oct 2024 08:53:45 GMT

Copyright © 2024, Versa Networks, Inc.


```
admin@versa-concerto:$ vsh system
Usage: vsh system [OPTIONS] COMMAND [ARGS]...
```

```
Options:
  --help Show this message and exit.
```

```
Commands:
  details
  eraseconfig
  osspack
  package
  echsupport
```

- **vsh system details**—Display basic build details about the installed Concerto software.

```
admin@versa-concerto:$ vsh system details
```

```
Software Details
```

```
-----
{"ver_major": "10",
 "ver_minor": "2",
 "ver_service": "1",
 "reltype": "",
 "date": "20210507",
 "time": "125603",
 "commit_id": "388274d33c7846815382a96559dffa255b08d918",
 "reflog_id": "388274d",
 "branch_name": "ecp-dev",
 "ecp_pkg": "concerto-20210507-125603-388274d-10.2.1",
 "build_type": "",
 "build_host": "",
 "built_by": ""}
-----
```

- **vsh system eraseconfig**—Erase the database and cluster information on all the nodes in the cluster. Running this command on any node brings down the entire Concerto cluster.
- **vsh system osspack info**—View the current OS Spack version installed on Concerto.

```
admin@versa-concerto:/opt/versa/ecp/scripts/utlis/cli$ vsh system osspack info
Version: 20210419
Update Type: full
```

- **vsh system package fetch url**—Download the package located at the indicated URL.

```
admin@versa-concerto:$ vsh system package fetch http://builds.versa-networks.com/concerto/ecp-dev/latest/concerto-20210421-010702-75d23e0-10.2.1.bin
=> Downloading Versa Concerto package ...
100% 100% => Package Saved.
```

- **vsh system package list**—List the packages currently available in the system.

```
admin@versa-concerto:$ vsh system package list
*.deb
```

```
concerto-20210421-010702-75d23e0-10.2.1.bin
```

- **vsh system package delete *filename***—Delete a package from the system.

```
admin@versa-concerto:$ vsh system package delete concerto-20210421-010702-75d23e0-10.2.1.bin
=> Removed concerto-20210421-010702-75d23e0-10.2.1.bin
```

- **vsh system package upgrade *filename.bin***—Upgrade a node to the software version of the specified *.bin file. You must run this command on every node in the Concerto cluster.
- **vsh system techsupport**—Capture technical support data so that you can debug system issues.

```
admin@versa-concerto:$ vsh system techsupport
Created /var/tmp/versa-ecp-support-dump-2021-Apr-20-185239.tgz
```

- **vsh cluster**—Display the **vsh cluster** commands and options.

```
admin@versa-concerto:$ vsh cluster
Usage: vsh cluster [OPTIONS] COMMAND [ARGS]...
```

```
Options:
  --help  Show this message and exit.
```

```
Commands:
  add-node
  eraseconfig
  info
  init
  modify
  regenerate-manifest
  remove-node
```

- **vsh cluster info**—Display the current status of a multinode deployment. Note that this command is available only for multimode deployments, not for single-node deployments.

```
admin@concerto-1:~$ vsh cluster info
```

```
Concerto Cluster Status
```

```
-----
Node Name:      concerto-1
IP Address:     10.48.7.81
Operational Status: primary
Configured Status: primary
Docker Node Status: ready
Node Reachability: reachable
GlusterFS Status: good
```

```
Node Name:      concerto-2
IP Address:     10.48.7.82
Operational Status: primary
Configured Status: primary
Docker Node Status: ready
Node Reachability: reachable
GlusterFS Status: good
```

```
Node Name:      concerto-3
IP Address:     10.40.30.80
Operational Status: primary
Configured Status: primary
Docker Node Status: ready
Node Reachability: reachable
GlusterFS Status: good
```

- **vsh cluster eraseconfig**—Erase the cluster-related data (reset the Docker swarm) and delete the system database.
- **vsh cluster init**—Create a multinode Concerto cluster deployment. A node can be deployed in one of three types: primary, secondary, and arbiter). When executing this command, you need to provide the IP address, node type, and user credentials for each node in the cluster. You also need to select the number of instances of the core Concerto services (the number of replicas of the services) that need to be deployed.

```
admin@concerto-1:~$ vsh cluster init
ARE YOU SURE, YOU WANT TO INITIALIZE THE ECP SETUP? OLD CONFIGS WILL BE DELETED - Y/N: y
Enter the number of nodes needed for the cluster: 3
-----
Enter the IP address for node 1: 10.48.7.81
Enter the username for node 1: admin
Enter the password for node 1:
Select the node type:
1. Primary
2. Secondary
3. Arbiter
Select the value: 1
-----
Enter the IP address for node 2: 10.48.7.82
Enter the username for node 2: admin
Enter the password for node 2:
Select the node type:
1. Primary
2. Secondary
3. Arbiter
Select the value: 1
-----
Enter the IP address for node 3: 10.40.30.80
Enter the username for node 3: admin
Enter the password for node 3:
Select the node type:
1. Primary
2. Secondary
3. Arbiter
Select the value: 1
Enter the number of replicas for core services (Default = 1 replica for each service), Hit Enter for default:
```

- **vsh cluster add-node**—Add additional nodes to a cluster.

```
admin@versa-concerto:$ vsh cluster add-node
Enter the number of nodes to be added in the cluster: 1
```

```
-----
Enter the IP address for node 1: 10.40.211.90
Select the node type:
1. Primary
2. Secondary
3. Arbiter
Select the value: 1
Enter the username for node 1: admin
Enter the password for node 1:
[ 10.40.211.90 ] : Executing cmd : hostname
```

```
[10.40.211.90] run: hostname
[10.40.211.90] out: ecp4
[10.40.211.90] out:
```

Enter the number of replicas for core services (Default = 1 replica for each service), Hit Enter for default:

- **vsh cluster rem-node**—Remove existing nodes from the cluster.

```
admin@versa-concerto:$ vsh cluster remove-node
Enter the number of nodes to be removed from the cluster: 1
-----
Enter the IP address for node 1: 10.40.211.90
Select the node type:
1. Primary
2. Secondary
3. Arbiter
Select the value: 1
Enter the username for node 1: admin
Enter the password for node 1:
[ 10.40.211.90 ] : Executing cmd : hostname

[10.40.211.90] run: hostname
[10.40.211.90] out: ecp4
[10.40.211.90] out:
```

- **vsh cluster modify**—Add or remove existing nodes from the cluster.

```
admin@versa-concerto:$ vsh cluster modify
Enter the number of nodes to be added in the cluster: 1
-----
Enter the IP address for node 1: 10.40.211.90
Select the node type:
1. Primary
2. Secondary
3. Arbiter
Select the value: 1
Enter the username for node 1: admin
Enter the password for node 1:
[ 10.40.211.90 ] : Executing cmd : hostname

[10.40.211.90] run: hostname
[10.40.211.90] out: SDWAN-VOAE1
[10.40.211.90] out:
```

Enter the number of nodes to be removed from the cluster: 1

Enter the IP address for node 1: 10.40.211.91

Select the node type:

1. Primary
2. Secondary
3. Arbiter

Select the value: 1

Enter the username for node 1: admin

Enter the password for node 1:

[10.40.211.91] : Executing cmd : hostname

[10.40.211.91] run: hostname

[10.40.211.91] out: SDWAN-VOAE2

[10.40.211.91] out:

Enter the number of replicas for core services (Default = 1 replica for each service), Hit Enter for default:

- **vsh cluster regenerate-manifest**—Regenerate Docker deployment manifest files. You can use this command when new builds contain changes in the Docker templates.
- **vsh start**—Start Concerto services. In a multinode deployment, you can run this command from any of the nodes.

Note: The no-timeout option was added in Releases 10.2.2 and later. It is primarily used for very slow setups that take more than the specified timeout values for zookeeper and postgres to start (provided there are no underlying issues or application errors). If zookeeper does not start within 10 minutes or if Postgres does not start within 20 minutes after executing the **vsh start** command, the command will not automatically exit.

```
admin@versa-concerto:$ vsh start no-timeout
```

```
=> Initializing docker swarm
=> Creating Docker Networks
=> Adding node label type as primary
=> Deploying Miscellaneous services...
=> Deploying Traefik loadbalancer...
=> Deploying Zookeeper and Kafka...
=> Waiting for zookeeper to start...
=> Deploying PostgreSQL service...
=> Waiting for postgres to start...
=> Deploying Flyway service...
=> Flyway migration succeeded
=> Deploying Hazelcast cache service...
=> Deploying Solr search service...
=> Deploying System Monitoring service...
=> Deploying Concerto services...
```

- **vsh stop**—Stop Concerto services. In a multinode deployment, you can run this command from any of the nodes.
- **vsh status**—View the status of Concerto services. In a multinode deployment, you can run this command from any of the nodes.

Supported Software Information

Releases 10.2.1 and later support all content described in this article, except:

- Releases 10.2.2 adds support for the no-timeout option with the **vsh start** command.

Additional Information

[Common Configuration](#)

[Monitor Analytics Clusters using Open Source Tools](#)