Turbonomic 6.0
Target Configuration Guide

Turbonomic, Inc.

500 Boylston St, 8th floor
Boston, MA 02116 USA
Phone: (866) 634 5087
www.turbonomic.com

COPYRIGHT

Copyright© Turbonomic 2010 — 2017, all rights reserved

END-USER LICENSE AGREEMENT

https://cdn.turbonomic.com/wp-content/uploads/Turbonomic_Click_Through_Customer-License.pdf

Table of Contents

Target Configuration

A target is a service that performs management in your virtual environment. Turbonomic uses targets to monitor workload and to execute actions in your environment. Target Configuration specifies the ports Turbonomic uses to connect with these services. You must install Turbonomic on a network that has access to the specific services you want to set up as targets.

For each target, Turbonomic communicates with the service via the management protocol that it exposes — The REST API, SMI-S, XML, or some other management transport. Turbonomic uses this communication to discover the managed entities, monitor resource utilization, and execute actions.

You can assign instances of the following technologies as Turbonomic targets:

- Hypervisors
    - Citrix XenServer 5.6.x and 6.x
    - IBM PowerVM
    - Microsoft Hyper-V 2008 R2, Hyper-V 2012, and Hyper-V 2012 R2
    - RHEV-M (RedHat Enterprise Virtualization Manager) versions 3.x
    - VMware vCenter 4.1 — 6.5 running with ESX 3.x, 4,x, 5.x, and 6.x
- Private Cloud Managers
    - CloudStack 4.0 — 4.6
    - Microsoft System Center 2012 Virtual Machine Manager and System Center 2012 R2 Virtual Machine Manager
    - VMware vCloud Director 1.0 — 5.1
    - OpenStack havana, Icehouse, juno, kilo, and Mitaka

- Public Cloud Services
    - Amazon AWS
        * us-east-1
        * us-east-2
        * us-west-1
        * us-west-2
        * ca-central-1
        * ap-south-1
        * ap-southeast-1
        * ap-southeast-2
        * ap-northeast-1
        * ap-northeast-2
        * eu-central-1
        * eu-west-1
        * eu-west-2
        * sa-east-1
    - Microsoft Azure
    - IBM SoftLayer
- Application Servers
    - IBM WebSphere Application Server, version 8.0.0.9 or greater
    - Oracle WebLogic versions 11g or 12c
    - JBoss Application Server 7.0 and later — JBoss Deployment Manager using jboss-eap-6.3
    - Apache Tomcat, versions 6.x, 7.x and 8.0.x
    - JVM Application
- Database Servers
    - Microsoft SQL Server 2008 R2, 2012, and 2014
    - Oracle 11g R2 and 12c
    - MySQL 5.5.26 and higher, and all 5.6 releases
- Guest OS Processes
    - WMI on Windows Server 2008R2 or later
    - SNMP with v2c connection
- Microsoft Applications
    - Microsoft Exchange
- Load Balancers
    - Citrix NetScaler
- Application Containers
    - Docker API 1.2 or greater
    - Kubernetes 1.4 or greater
- Storage Managers
    - NetApp Storage Systems running Data ONTAP version 8.x or 9.0
    - EMC VMAX 2 and 3
    - EMC VNX Series Storage Systems (for version details, see the EMC VNX Support KB article)
    - EMC VPLEX
    - EMC XtremIO
    - Pure Storage FlashArray
    - HP 3PAR StoreServ
    - Nutanix
    - Dell Compellent

- Fabric Managers
  - Cisco UCS 2.0 and higher or UCS C-Series Rack Server
  - Cisco UCS Central
  - HPE OneView
- Network Flow Collectors
  - NetFlow/sFlow: NFDUMP — Turbonomic provides an OVA download with NFDUMP preconfigured for NetFlow and sFlow collection
  - Arista EOS+
  - Cisco APIC
- Turbonomic Targets
  To configure an aggregated deployment of Turbonomic, you can assign Turbonomic servers as targets. The versions of target instances must match the version of the aggregating instance.

Adding and Removing Target Virtual Management Servers

The target services your Turbonomic installation will manage appear in the Target Configuration list. You can add, remove, and edit entries in this list. Note that the target service's account must be configured with privileges that support the Turbonomic activities you want to perform. For example, the following list shows how vCenter privileges correspond to activities Turbonomic can perform:

- **Read Only** — Enables Turbonomic monitoring and simulation (what-if scenarios) only
- **VCenter Administrator** — Enables Turbonomic monitoring, simulation (what-if scenarios), and automation functions
- **Enable Datastore Browse** — Enabling this property for the account gives Turbonomic the privileges it needs to enable its storage management functionality

Adding Targets

To add a target service, click the **Target Configuration** button, provide the requested information, and click **Apply** to validate those targets and start a new discovery.

Typical information you provide includes:

- Target Type — Choose among the supported VM Management technologies (Hypervisor, Cloud Management, Load Balancer, etc.)
  After you choose the technology, then choose the specific target type for that technology. For example, for Hypervisor technology, the types you can choose include vCenter, RHEV, Hyper-V, and XenServer.
- Hostname or IP address — The address of the target service you want to add
- User Name — A valid account username for the target service
- Password — A password for the target service account

Editing and Removing Targets

To edit a target entry, select it in the list and then click **Edit**. The **Target Configuration Form** opens, where you can make your settings.

To remove a target, select the entry in the list and then click **Delete**.

Hypervisor Targets

A hypervisor is a service that creates and runs virtual machines (VMs), providing the VMs compute and storage resources. When you connect Turbonomic to hypervisor targets in your environment, Turbonomic controls your environment, assuring application performance while also utilizing resources as efficiently as possible.

One of the first steps in any Turbonomic deployment is to connect to the hypervisors within your environment. Once connected, Turbonomic discovers the VMs, the physical machines that host the VMs, the datastores that provide storage resources to the physical machines, and the virtual datastores that provide storage resources to the VMs.

Supply Chain

Each hypervisor requires a physical machine (host) and one or more datastores to provide compute and storage resources. Virtual machines (VMs) run on those physical resources, and the VMs in turn provide resources to applications.

At the bottom of the supply chain, physical machines consume resources from data centers.

If your environment includes SAN technologies such as disk arrays, then the storage consumes resources from that underlying technology. If you add these storage targets, then Turbonomic extends the supply chain analysis into the components that make up the disk array. For more information, see Storage Targets on page 66.

Actions

Turbonomic recommends actions for the hypervisor supply chain as follows.

NOTE: This is a general list of actions for entities discovered for hypervisors. Detailed actions per target are described in each target section.

Entity Type	Action
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Physical Machines	• Start Physical Machine • Provision Physical Machine • Suspend Physical Machine
Storage	• Start Storage • Provision Storage • Suspend Storage • Move (only with Storage Targets configured) • Resize (only with Storage Targets configured)
Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC

Monitored Resources

Turbonomic monitors the following resources for the hypervisor supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB)

	• CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Datacenter	***NOTE:*** *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.* • Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s)

	• Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Kilobytes (KB)
Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)

IBM PowerVM

Turbonomic connects with IBM PowerVM servers through the IBM Hardware Management Console (HMC) to manage logical partitions (LPARs), virtual I/O servers (VIOS), and the associated network and storage.

The components of IBM PowerVM map to entities in the Turbonomic supply chain as follows:

PowerVM Name	Turbonomic Entity
Frame	Physical Machine
Node	N/A — A source of compute resources contained within the frame, but not described in the Turbonomic user interface as a separate entity
LPAR or VIOS	Virtual Machine
Fabric	Storage

Prerequisites

- IBM Hardware Management Console (HMC) 8.3 or higher with the Performance and Capacity Monitoring API and data collection enabled.
- A service account Turbonomic can use to connect to your HMC. To generate actions, the account must have HMC Viewer permission and access to the HMC Capacity and Performance Monitoring API. To automate actions the account must have HMC Root permissions.

NOTE: To discover utilization (Mem, CPU, etc.) on HMC frames, the account's user permissions must include the `List Utilization Data` permission for each frame.

- To accurately monitor LPAR memory, and to recommend actions based on LPAR memory utilization:
  - Each LPAR must use an AIX operating system
  - Each LPAR must have SNMP enabled

  When you specify an HMC target, you specify an SNMP port and a community name — The values you provide for the target should match the SNMP configuration on the LPARs that the HMC manages.

  If you do not enable SNMP connections to a given LPAR, then Turbonomic successfully manages that LPAR but it cannot read memory utilization from the LPAR. Without SNMP to an LPAR running AIX, Turbonomic gets memory consumption values from the Hardware Management Console. HMC memory values are not based on actual utilization in the LPAR.

Adding PowerVM Targets

To add PowerVM targets, select the **Hypervisors > IBM PowerVM** option on the Target Configuration page and provide the following information:

- Address
  The name or IP address of the Hardware Management Console.
- Username/Password
  Credentials for the service account Turbonomic can use to connect to the HMC.

- SNMP Port
  An SNMP port common to the LPARs managed by the target HMC.
- SNMP Community
  The SNMP community common to the LPARs managed by the target HMC.

Resizing Down Virtual I/O Servers

If you do not want to receive "Resize down" recommendations for your virtual I/O servers, you can disable this on the Policy tab as follows:

- In the Policy Editor, select **Action > VM**
- In the Scope pane, expand **Folders** and select **AIX-vioses-auto**
- For the **Resize down** action in the Action Mode Settings table, enable **Override** and change the value to **Disabled**
- Click **Apply Settings Change**

For more information about Policies, see the Turbonomic User Guide.

Supported Actions

Turbonomic supports the following actions for IBM PowerVM entities:

Entity Type	Can Be Automated	Recommendations Only
Virtual Machine	Move, Resize Down, Resize Up	Start, Suspend, Terminate, Provision, Reconfigure
Physical Machine		Start, Suspend, Terminate, Provision

Monitored Resources

Turbonomic monitors the following resources for the hypervisor supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS

	• Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)

Datacenter

NOTE: *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.*

- Memory (Mem)
  The utilization of the PM's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the PM's CPU reserved or in use
  Measured in Megahertz (MHz)
- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared of memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Kilobytes (KB)

Hyper-V

If you have a small number of Hyper-V hosts in your environment, you can add them individually as Turbonomic targets. Also, if you have deployed the Hyper-V hosts in a clustered domain (for example as a failover cluster), you can specify one Hyper-V host as a target and Turbonomic automatically add the other members of that cluster.

Note that for large Hyper-V environments, it’s typical to manage the hosts via System Center Virtual Machine Manager (VMM). You can specify the VMM server as a target and Turbonomic will use it to discover and manage its child Hyper-V hosts. If you use VMM, you should not add individual Hyper-V hosts as targets. For information about adding VMM targets, see Virtual Machine Manager on page 48.

> **NOTE:** Even if you manage Hyper-V using a VMM target, you must still configure remote management on each Hyper-V server. This Hyper-V topic includes instructions to configure remote management — see Creating A Service User Account on page 17.

For accurate SMB storage calculations, Turbonomic requires a VMM target.

Prerequisites

- Create a service account that Turbonomic can use to connect to your Hyper-V servers. See "Creating a Service User Account in Hyper-V" in the Turbonomic User Guide
- Configure remote management on each Hyper-V server. Refer to the relevant section below:
  - **Turbonomic** 5.4 or higher: See Creating A Service User Account on page 17
  - **Turbonomic** 5.3 or below: See Enabling Management via DCOM on page 189

Adding Hyper-V Targets

Once you've enabled remote management, you can add your Hyper-V hosts as targets. To add Hyper-V targets, select the **Hypervisors > Hyper-V** option on the Target Configuration page and provide the following information:

- Address

  The FQDN of the Hyper-V host. If you're using the "Discover Host Cluster" below to add an entire cluster, enter the name of any one of the Hyper-V hosts in the cluster.

  Note that you can enter an IP address for the host, but you must first configure an SPN on the host. Turbonomic recommends that you use the FQDN in this field.
- Port number

  The port number for the remote management connection. The default HTTP port is 5985; the default HTTPS port is 5986.
- Secure connection

  Select this option to use a secure connection (HTTPS). Make sure the required certificate is configured for use on the host.
- Full domain name

  The full domain name of the cluster to which the host belongs.
- Discover Host Cluster

  Turbonomic discovers and adds all Hyper-V hosts in the named cluster if this option is checked. Note that each server must be configured to allow remote management. You may find it helpful to configure WinRM using a GPO so new servers are configured automatically (see Enabling WinRM Via Global Policy Objects on page 191).
- Username

  The username of a service account Turbonomic can use to connect to the Hyper-V host. If you checked "Discover Host Cluster" in the field above, use an account that is valid for all Hyper-V hosts in that cluster.

  For Turbonomic 5.3 or below: The account must also be one of the server's WBEM Scripting Locator owners (see Enabling Management via DCOM on page 189).
- Password

  Password for account used.

NOTE: If your Hyper-V hosts are running in a Nutanix environment, you must understand pinning a Nutanix Controller VM. For more information, see Controller VM Pinning on page 93.

Hyper-V SDK Probe Properties

Turbonomic offers the capability to adjust certain properties of the probe. In general, this tweaking will not be necessary. If adjustment is needed, the file can be found in the `/srv/tomcat/webapps/vmturbo/probe-jars/{probe}/resources` folder. This target has the following properties:

Setting	Function
maxEnvelopeSize	Maximum size of SOAP response envelope
timeout	Determines the timeout in milliseconds for both initial connection and discovery, and maximum period between data packets
conversionObjectsBatchSize	Maximum amount of internal objects that should be converted into EntityDTO in one batch
queriesThreadPoolCoreSize	Number of threads that will remain in the thread pool even if idle timeout is exceeded. This number should be lower or equal to `queriesThreadPoolMaximumSize`
queriesThreadPoolMaximumSize	Maximum number of threads that will be used by the thread pool
queriesThreadPoolIdleTimeout	Time in seconds before a thread will time out if number of active threads are above `queriesThreadPoolCoreSize`
hostThreadPoolCoreSize	Number of threads that will remain in the thread pool responsible for simultaneous discovery of host instances. This number should be lower or equal to `hostThreadPoolMaximumSize`
hostThreadPoolMaximumSize	Maximum number of threads that will be used by the thread pool responsible for simultaneous discovery of host instances
hostThreadPoolIdleTimeout	Time in seconds before a thread will time out if number of active threads are above `hostThreadPoolCoreSize`
clusterThreadPoolCoreSize	Number of threads that will remain in the thread pool for handling requests to Hyper-V targets. This number should be lower or equal to `clusterThreadPoolMaximumSize`
clusterThreadPoolMaximumSize	Maximum number of threads that will be used by the thread pool responsible for handling requests to Hyper-V hosts

clusterThreadPoolIdleTimeout	Time in seconds before a thread will time out if number of active threads are above `clusterThreadPoolCoreSize`
hypervCommonThreadPoolCoreSize	Number of threads that will remain in the thread pool responsible for miscellaneous SDK tasks
hypervCommonThreadPoolMaximumSize	Maximum number of threads that will be used by the thread pool responsible for miscellaneous SDK tasks. This number should be lower or equal to `hypervCommonThreadPoolMaximumSize`
hypervCommonThreadPoolIdleTimeout	Time in seconds before a thread will time out if number of active threads are above `hypervCommonThreadPoolCoreSize`
actionStepCheckSucceededTimeout	Time in milliseconds to wait before triggering the next step in an action chain
actionStepCheckAmount	Number of times to poll for an action is complete before failure
actionStepCheckTimeout	Time in milliseconds between action checks
requestsPerHost	Maximum number of simultaneous requests per host

NOTE: To improve the performance of discovery in large environments (25+ Hosts), increase the following settings:

- `queriesThreadPoolMaximumSize`
- `hostThreadPoolMaximumSize`
- `clusterThreadPoolMaximumSize`

This increases the number of threads that the probe uses to discover your environment. As you increase the number of threads, you may need to increase the amount of memory allocated to Turbonomic.

Supported Actions

For each discovered entity within the hypervisor supply chain, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations Only
Virtual Machine	Start, Move, Suspend, Storage Move, Resize Down, Resize Up	Terminate, Provision, Reconfigure
Physical Machine	Start, Suspend	Terminate, Provision
Storage		Provision

Monitored Resources

Turbonomic monitors the following resources for the hypervisor supply chain:

Entity Type	Commodity
VirtualMachine	Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB)
	Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz)
	Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB)
	Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS
	Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB)
	CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz)
	IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s)
	Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s)
	Swap The utilization of the PM's swap space Measured in Kilobytes (KB)
Storage	Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB)
	Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB)

- Storage Access Operations Per Second (IOPS)
  The utilization of the storage access operations per second on the datastore
  Measured in Operations per second
- Latency
  The utilization of latency on the datastore
  Measured in Milliseconds (ms)

Datacenter

NOTE: *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.*

- Memory (Mem)
  The utilization of the PM's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the PM's CPU reserved or in use
  Measured in Megahertz (MHz)
- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared of memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Kilobytes (KB)

Creating A Service User Account

The service account Turbonomic uses to connect to a Hyper-V host must be an Active Directory domain account. The account must have full access to the cluster. To create such an account, execute the following command at a PowerShell prompt:

```
Grant-ClusterAccess <domain>\<service_account> -Full
```

Additionally, the service account must have specific local access rights on each host. The easiest way to grant Operations Manager the access it requires is to add the domain account to the `Local Administrators` group on each Hyper-V server.

Some enterprises require that the service account does not grant full administrator rights. In that case, you can create a restricted service account on every Hyper-V host:

1. **Add the service account to each of the following local groups:**
   - `WinRMRemoteWMIUsers__` (or `Remote Management Users`)
   - `Hyper-V Administrators`
   - `Performance Monitor Users`

NOTE: These groups are standard Windows Server 2012 security groups. If you are using an earlier version of Windows Server and do not see these groups, contact Turbonomic Support for assistance.

2. **Grant permissions to the service account.**

   In the WMI Management console, grant the `Enable Account` and `Remote Enable` advanced security permissions to the service account:
   - Open the WMI Management console (`wmimgmt`).
   - Right-click **WMI Control (Local)** and choose **Properties**.
   - Go to the **Security** tab and then click **Security** to display the **Security for Root** dialog.
   - Click **Advanced**, select the service account, and click **Edit**.
   - Confirm that **This namespace and subnamespace** is selected.
   - Select **Enable Account** and **Remote Enable** and click **OK**.
3. **Configure the WinRM security descriptor to allow access by the service account:**
   - At a PowerShell prompt, execute `winrm configSDDL default`.
   - In the "Permissions for Default" dialog box, grant the service account Read and Execute access.

Red Hat Enterprise Virtualization (RHEV-M)

Red Hat Enterprise Virtualization Manager (RHEV-M) provides centralized management of Red Hat hypervisors. To manage Red Hat hypervisors, you specify a RHEV-M instance as a target.

Prerequisites

- A service account Turbonomic can use to connect to the RHEV-M server. The account must have administrator privileges on RHEV-M.

Adding RHEV-M Targets

To add RHEV-M targets, select the **Hypervisors > RHEV-M** option on the Target Configuration page and provide the following information:

- Address

  The name or IP address of the RHEV-M console. Turbonomic defaults to port 443. If your RHEV-M uses a port other than 443, append the port number to the address. For example, RHEV-M 3.0 uses port 8443 — You would specify the following address:`<RHEV-M_IP_ADDRESS>:8443`.

- Username/Password
  Credentials for the service account Turbonomic can use to connect to RHEV-M. If you’re connecting using an account that is not the internal `admin` account, specify `<domain>\<username>` for the username.

Supported Actions

For each discovered entity within the hypervisor supply chain, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations Only
Virtual Machine	Start, Move, Suspend, Storage Move, Resize Down, Resize Up	Terminate, Provision, Reconfigure
Physical Machine	Start, Suspend	Terminate, Provision
Datastore ("Storage")		Provision

Monitored Resources

Turbonomic monitors the following resources for the hypervisor supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz)

	• IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Datacenter	***NOTE:*** *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.* • Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB)

	• Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB)
	• CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Kilobytes (KB)
Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB)
	• CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz)
	• Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB)
	• CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz)
	• Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)

vCenter Server

VMWare vCenter Server provides a centralized management platform for VMware hypervisors. To manage your VMware environment with Turbonomic, you specify a vCenter Server instance as a target. Turbonomic discovers the associated infrastructure and performs intelligent workload management.

Prerequisites

- A service user account Turbonomic can use to connect to your vCenter and execute actions (see Creating A Service User Account In vCenter on page 25).

General Considerations Before Target Addition

- **Linked vCenters**:
  For linked vCenters, you must add each vCenter separately. Turbonomic communicates with each vCenter through a separate API endpoint, but aggregates the data from those vCenters and makes cross-vCenter actions possible.
- **Shared Datastores**:
  If you add more than one vCenter target that manages the same datastore, you may see conflicts in the metadata Turbonomic maintains for each vCenter. For example, datastore browsing may display a conflict between active and wasted files, or each vCenter may define the same datastore as a member of a different storage cluster. Turbonomic recommends that you do not add multiple vCenter targets that manage the same datastore.
- Restricting **Turbonomic** Access to Specific Clusters:
  When you add a vCenter target, Turbonomic discovers all of the connected entities that are visible based on the account you're using to connect to the vCenter. If there are clusters or other entities you want to exclude, you can do this by setting the role for the Turbonomic account to `No access` in the vSphere management client.

Adding vCenter Targets

To add vCenter targets, select the **Hypervisors > vCenter** option on the Target Configuration page and provide the following information:

- Address
  The name or IP address of the vCenter server.
- Username/Password
  Credentials for the service account Turbonomic can use to connect to the vCenter Server. Include the domain if required (<*domain*>\<*username*>).

NOTE: If your VMware hypervisors are running in a Nutanix environment, you must understand pinning a Nutanix Controller VM. For more information, see Controller VM Pinning on page 93.

vCenter Post-Addition Imports

In addition to discovering entities managed by the hypervisor, Turbonomic also imports a wide range of vSphere settings, such as Host DRS rules, annotations, Resource Pools, and DRS HA settings.

NOTE: *Turbonomic does not import Storage DRS settings.*

VMware vSphere 6.0 introduced the ability to move VMs between vCenters. If you enabled this feature in your VMware environment, you can configure Turbonomic to include cross vCenter vMotions in its recommendations.

To configure Operations Manager to support cross vCenter vMotion recommendations, you must create a Workload Placement Segment that merges the clusters on the different vCenters. To create a merge segment:

- On the **Policy > Workload Placement** page, click the **Add Segment** button.
- In the Segment Editor window, select the **Merge** option and add the vCenter clusters to merge.
  For more information about Policies, see the Turbonomic User Guide.

NOTE: Since Turbonomic can only execute vMotions between clusters that use the same switch type (VSS or VDS), make sure any clusters you merge use the same switch type. Although Turbonomic will not initiate VSS → VDS vMotions, vSphere may do so. If this happens, Turbonomic displays a compliance violation notification.

Supported Actions

For each discovered entity within the hypervisor supply chain, Operations Manager can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations Only
Virtual Machine	Start, Move, Suspend, Storage Move, Resize Down, Resize Up	Terminate, Provision, Reconfigure
Physical Machine	Start, Suspend	Terminate, Provision
Storage		Provision

Monitored Resources

Turbonomic monitors the following resources for the hypervisor supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz)

- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Megahertz (MHz)

Storage

- Storage Amount
  The utilization of the datastore's capacity
  Measured in Megabytes (MB)
- Storage Provisioned
  The utilization of the datastore's capacity, including overprovisioning.
  Measured in Megabytes (MB)
- Storage Access Operations Per Second (IOPS)
  The utilization of the storage access operations per second on the datastore
  Measured in Operations per second
- Latency
  The utilization of latency on the datastore
  Measured in Milliseconds (ms)

Datacenter

NOTE: *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.*

- Memory (Mem)
  The utilization of the PM's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the PM's CPU reserved or in use
  Measured in Megahertz (MHz)
- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)

	• Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Kilobytes (KB)
Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)

Creating A Service User Account In vCenter

The service account you use must have specific permissions on the vCenter. The easiest way to grant Turbonomic the access it requires is to grant full administrator rights.

Some enterprises require that the service account does not grant full administrator rights. In that case, you can create a restricted service account that grants the following permissions to enable the required Turbonomic activities:

Turbonomic Activity	Required Privileges
Monitoring	Read-only for all entity types
Recommend Actions	Read-only for all entity types
Wasted Storage Reporting	Datastore > Browse Datastore
Execute VM Move (vMotion)	Resource > Migrate Resource > Query Vmotion Resource > Modify Resource Pool Resource > Assign VM to Resource Pool

Turbonomic Activity	Required Privileges
Execute VM Storage Move (svMotion)	Datastore > Allocate Space Datastore > Browse Datastore Datastore > Configure Datastore Datastore > Move Datastore Datastore > Remove File Datastore > Update Virtual Machine Files Datastore Cluster > Configure a Datastore Cluster Resource > Assign VM to Resource Pool Resource > Migrate Resource > Relocate Resource > Modify Resource Pool Resource > Move Resource Pool Resource > Query VMotion Virtual Machine > Configuration > Change Resource Virtual Machine > Configuration > Swap File Placement **NOTE**: Datastore Cluster permissions only apply to vSphere 5.x
Execute VM Resize	Virtual Machine > Configuration > Change CPU Count Virtual Machine > Configuration > Change Resources Virtual Machine > Configuration > Memory Virtual Machine > Interaction > Reset Virtual Machine > Interaction > Power Off Virtual Machine > Interaction > Power On
Discover tags	Global > Global tags In addition, you must open ports 10443 and 7443 on the target server

XenServer

XenServer is a virtualization platform that provides centralized management of virtual resources. The XenServer environment configures hosts as resource pools with shared storage. A resource pool always has at least one physical node that serves as the resource pool master. The master exposes an administration interface, and forwards commands to other members of the pool. Turbonomic connects to a pool through the resource pool master — you specify the resource pool master as a target.

Prerequisites

- A service account Turbonomic can use to connect to the XenServer host that serves as a resource pool master. In addition, this account must have administrator privileges on all hosts in the resource pool.
- For full integration, XenServer 6.1 or higher with the Performance Monitoring Enhancements Pack is required. Earlier versions of the API do not provide Turbonomic with storage IOPS and latency information.
- All hosts in the environment must have their clocks synchronized via NTP. Turbonomic is unable to collect utilization data from XenServer hosts if they are not time synchronized.

Adding XenServer Targets

To add XenServer targets, select the **Hypervisors > XenServer** option on the Target Configuration page and provide the following information:

- Address
  The name or IP address of the Resource Pool Master.
- Username/Password
  Credentials for the service account Turbonomic can use to connect to the Resource Pool Master.

Supported Actions

Entity Type	Can Be Automated	Recommendations Only
Virtual Machine	Start, Move, Suspend, Resize Down, Resize Up	Terminate, Provision, Storage Move, Reconfigure
Physical Machine		Start, Suspend, Terminate, Provision
Datastore ("Storage")		Provision

Monitored Resources

Turbonomic monitors the following resources for the hypervisor supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB)

	• CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Datacenter	***NOTE:*** *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.* • Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s)

- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared of memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Kilobytes (KB)

Cloud Management Targets

Cloud-based datacenters support scalability, resource pooling, multi-tenancy, and self-service management of virtual resources. Turbonomic supports the following cloud technologies:

- Private cloud
    - Apache CloudStack
    - Microsoft Virtual Machine Manager (VMM)
    - OpenStack Cloud Operating System
    - VMware vCenter Server
- Public cloud
    - Amazon Web Services (AWS)
    - IBM SoftLayer
    - Microsoft Azure
- Hybrid clouds
  A combination of private and public clouds — Turbonomic can burst workload out to the public cloud to accommodate peaks in demand. For more information about this configuration, see "Hybrid Cloud Configuration" in the Turbonomic User Guide.

Turbonomic can manage private and public clouds, ensuring that applications running on the cloud have enough resources to assure performance. For example, if applications on private cloud need more resources, Turbonomic can recommend that the self-service customer add more virtual resources. Or if necessary, Turbonomic can provision more physical machines to increase the pool of resources available to all the self-service customers.

Cloud infrastructures declare resource pools to serve business needs for multi-tenancy, authorized visibility into the infrastructure, and regional distribution of resources. These pools manage resources such as compute, storage, memory, and network bandwidth in ways that support the cloud's business requirements. Turbonomic discovers and manages these resource pools in relation to the whole datacenter. This enables Turbonomic to:

- Display cloud resources within each pool
- Analyze workload distribution across the entire datacenter
- Control the distribution of workload at each level of the cloud infrastructure

Actions

Turbonomic recommends actions for the cloud target supply chain as follows.

NOTE: This is a general list of actions for entities discovered for cloud targets. You can see how actions differ per technology in each section that describes adding a specific type of cloud target.

Entity Type	Action
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Physical Machines	• Start Physical Machine • Provision Physical Machine • Suspend Physical Machine
Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

NOTE: This is a general list of commodities for entities discovered for application servers. You can see how actions differ per technology in each section that describes adding a specific type of application server target.

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Physical Machine

- Memory (Mem)
  The utilization of the PM's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the PM's CPU reserved or in use
  Measured in Megahertz (MHz)
- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Megahertz (MHz)

Datacenter

NOTE: *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.*

- Memory (Mem)
  The utilization of the PM's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the PM's CPU reserved or in use
  Measured in Megahertz (MHz)
- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared of memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)

	• CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Kilobytes (KB)
Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)
Zone	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)

Region

- Memory (Mem)
  The utilization of the PM's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the PM's CPU reserved or in use
  Measured in Megahertz (MHz)
- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared of memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Megahertz (MHz)

Private Cloud

A private cloud manages resources in pools to support multi-tenancy and self-service provisioning of virtual workloads. Turbonomic manages these resource pools in real time as demand fluctuates. This includes demand across resource pools, virtual datacenters (VDCs), and tenants.

On the private cloud, you can use Turbonomic to:

- Set up charge-back and show-back for private cloud or service-provider scenarios
- For service-providers, set up scoped views to limit exposure to the customer base
- Plan hardware requirements — the planning scenarios takes cloud architectures into account

Supply Chain

For private clouds, Turbonomic discovers resource partitions that are managed by the cloud manager, as well as the workload running on these partitions (the VMs and applications) and, where applicable, the supply that hosts workload (the physical machines and storage). Turbonomic represents these partitions as the following types of Virtual Datacenters (VDCs):

- Provider VDC
  A collection of physical resources (PMs and datastores) within a private cloud. The cloud administrator has access to these resources, and defines the datacenter members. Administrators allocate Provider VDCs to manage

resources that will be allocated to external customers through one or more Consumer VDCs.

- Consumer VDC
  A collection of resources that are available for customers to perform self-service management of workload through the cloud. It is an environment customers can use to store, deploy, and operate virtual systems. Consumer VDCs use the resources supplied by a Provider VDC.

NOTE: Different targets use different names to refer to Virtual Datacenters. In the Turbonomic supply chain, these entities are all represented by Consumer and Provider VDCs, as follows:

Turbonomic	vCloud Director	vCenter Server	VMM	CloudStack	OpenStack
Consumer VDC	Organization VDC	Resource Pool (Child)	Tenant or TenantQuota	Accounts	Tenant
Provider VDC	Provider VDC	Resource Pool (Root)	Cloud	Pod	N/A

Actions

Turbonomic recommends actions for private cloud infrastructures as follows:

Entity Type	Action
Provider Virtual Datacenters	Turbonomic does not recommend actions to perform on a Provider VDC. Instead, it recommends actions to perform on the devices running in the datacenter.
Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC

Monitored Resources

Turbonomic monitors the following private cloud infrastructure resources:

Entity Type	Commodity
Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB)

- CPU
  The utilization of the Datacenter's CPU reserved or in use
  Measured in Megahertz (MHz)
- Storage
  The utilization of the storage attached to the Consumer vDC.
  Measured in Kilobytes (KB)

CloudStack

Apache CloudStack deploys and manages VMs and storage on a multi-tenancy infrastructure. Turbonomic supports CloudStack servers as targets, in an environment where CloudStack manages the hypervisors that you have specified as Turbonomic targets. For information about hypervisor targets, see Hypervisor Targets on page 5.

NOTE: After setting a CloudStack server as a target, you must then set the hypervisors that server will manage as targets. If you want Turbonomic to manage CloudStack without specifying the hypervisor targets as well, please contact technical support.

Adding CloudStack Targets

To add CloudStack targets, select the **Cloud Management > CloudStack** option on the Target Configuration page, and provide the following information:

- Host Name or IP Address
  Name or IP of the CloudStack Management Server, or the primary server in a Management Server Cluster.
  Turbonomic assumes port 443 by default. If your system communicates over a different port, specify that port in the address field.
  For example: `10.10.123.45:444`
- Username and Password
  The credentials must be for a Root Administrator user on the Management Server. In addition, this user must have a generated pair of API and Secret keys for the CloudStack API. Turbonomic uses the credentials you provide here to log into the CloudStack server and discover the user’s key values.

Other Information Imported from CloudStack

CloudStack supports Affinity Groups to express rules about VM placement — Currently CloudStack supports anti-affinity rules. Turbonomic discovers these Affinity Groups and imports them as Workload Placement Policies (see "Importing Workload Placement Policies" in the Turbonomic User Guide). Imported rules are enabled by default, but you can disable specific rules if you want.

Actions

Turbonomic recommends the following actions for Cloudstack targets:

Entity Type	Action
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Physical Machines	• Start Physical Machine • Provision Physical Machine • Suspend Physical Machine
Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC

Monitored Resources

Turbonomic monitors the following resources for Cloudstack targets:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s)

	• Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Datacenter	***NOTE:*** *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.* • Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Kilobytes (KB)
Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz)

	• Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)
DPod	• Memory (Mem) The utilization of the DPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the DPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the DPod Measured in Kilobytes (KB) • Flow The utilization of the network flow capacity utilized by the DPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization Measured in Kilobytes per second (KB/s)

OpenStack

To manage your OpenStack® environment, Turbonomic connects to the Keystone identity service endpoint. Through this connection, Turbonomic discovers the other services it needs to control your OpenStack environment. If it discovers the necessary services, then it considers the OpenStack target to be validated.

Turbonomic has been tested with hypervisors running Openstack on Red Hat Enterprise Linux and Red Hat RDO.

Prerequisites

Turbonomic supports OpenStack version Icehouse or later.

You must have the following services installed and enabled, and Turbonomic must have access to each service at its administrative endpoint:

- Keystone — Identity service, to support discovery of OpenStack tenants
- Nova — Host and manage cloud computing systems (hypervisors)
- Cinder — Manage block storage (Cinder volumes)
- Ceilometer — Collect and persist data on the utilization of physical and virtual resources (telemetry)
- Gnocchi (Optional) — Collect and persist data on the utilization of physical and virtual resources via a metrics and resources database.

  Turbonomic recommends using Gnocchi in addition to ceilometer for more consistent metric collection.

To verify that these services are enabled, Open the Horizon dashboard to the Admin/System page, or execute the following command at the OpenStack command line:

```
[root@openstack ~(keystone_admin)]# keystone service-list
```

You should see each of the required services in the list.

Adding OpenStack Targets

To add OpenStack targets, select **Cloud Management > OpenStack** on the Target Configuration page, and provide the following information:

- Hostname or IP Address
  Provide the public URL of the Keystone service. The default port is 5000 — Do not provide a port if you want to use the default. For the default port, validation assumes a standard HTTP connection. If you provide a port value (for a port other than 5000), validation assumes a secure HTTPS connection unless you specify the protocol.
  For example, `10.10.123.45:5001` will use HTTPS, while `http://10.10.123.45:5775` will use HTTP.
- Username
  The account must have an administrator role on the specified tenant. This account must be authenticated by OpenStack.
- Password
  The password for the administrator account.
- Tenant Name
  The organizational structure within the Compute service that you want to manage. In a basic OpenStack installation this tenant is usually named `admin`.

Enabling Reservations

When you add a valid OpenStack target, Turbonomic can perform its analysis, recommend actions, and perform actions to assure performance and efficiency in your environment. To support these actions, you do not need to perform other configuration.

Turbonomic also includes reservations — deployment capabilities that can act as workload orchestration, or be integrated into an existing orchestration system (see the Deploy View in the Turbonomic user interface). These capabilities include:

- Calculate optimal placement for new workload
- Reserve resources for proposed workload, and include the reservations in real-time and planning analysis

An OpenStack user can use the Turbonomic placement proposals to deploy workload to the optimal locations.

To enable these capabilities, you must install the Turbonomic Nova Scheduler plugin that matches your version of OpenStack. Turbonomic delivers the following versions of this plugin on the Turbonomic Github repository:

- Icehouse
  https://raw.githubusercontent.com/turbonomic/nova/stable/icehouse/nova/scheduler/vmt_scheduler.py
- Juno
  https://raw.githubusercontent.com/turbonomic/nova/stable/juno/nova/scheduler/vmt_scheduler.py
- Kilo
  https://raw.githubusercontent.com/turbonomic/nova/blob/stable/kilo/nova/scheduler/vmt_scheduler.py

- Mitaka
  https://raw.githubusercontent.com/turbonomic/nova/stable/mitaka/nova/scheduler/vmt_scheduler.py

To fetch the scheduler plugin you want, execute the following commands on the Nova controller (substituting the url to the version of scheduler plugin that you want):

```
cd /usr/lib/python2.6/site-packages/nova/scheduler/
curl -O <URL TO CORRECT VERSION OF vmt_scheduler.py>
```

After you execute these commands to add the plugin to your controller, add the following entries to the file, `/etc/nova/nova.conf` under the `[DEFAULT]` section, where you provide the IP address of your Turbonomic server, and credentials for a Turbonomic user account that has administrator privileges:

```
scheduler_driver = nova.scheduler.vmt_scheduler.VMTScheduler
vmturbo_rest_uri = <VMTurbo_IPAddress>
vmturbo_username = <VMTurbo_UserName>
vmturbo_password = <VMTurbo_Password>
```

After you restart the Nova scheduler, it can use the plugin to communicate with your Turbonomic instance.

Collecting Physical Machine Metrics

To collect full metrics from physical machines in your environment and display them in Turbonomic, the following actions must be taken:

- `/etc/ceilometer/pipeline.yaml` must be modified to match the following snippets:

```
  - name: meter_snmp
          interval: 600
          resources:
          - [Compute Node FQDN]
          - [Compute Node FQDN]
          - [Compute Node FQDN]

          meters:
          - "hardware.cpu*"
          - "hardware.memory.used"
          - "hardware.disk.size.used"
          sinks:
          - meter_sink
          - name: meter_snmp_rate
          interval: 600
          resources:
          - [Compute Node FQDN]
          - [Compute Node FQDN]
          - [Compute Node FQDN]
          meters:
          - "hardware.network*"
          - "hardware.system_stats*"
          sinks:
          - snmp_sink
 - name: snmp_sink
          transformers:
          - name: "rate_of_change"
          parameters:
          source:
```

```
          map_from:
          name:
"hardware\\.(system_stats.io|network)\\.(incoming|outgoing)\\.(blocks|bytes)"
          unit: "(blocks|B)"
          target:
          map_to:
          name: "hardware.\\1.\\2.\\3.rate"
          unit: "\\1/s"
          type: "gauge"
          publishers:
          - notifier://
```

- After modifying this file, you will need to create a `snmpd.conf` file. For information about creating this file, contact Technical Support.
  The following Green Circle article includes instructions for this procedure: OpenStack Compute Node Changes to Enable PM Metric Collection.

After these modifications are made, OpenStack will collect all available metrics for physical machines.

Actions

Turbonomic recommends actions for the cloud target supply chain as follows.

Entity Type	Action
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Physical Machines	• Start Physical Machine • Provision Physical Machine • Suspend Physical Machine
Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC

Monitored Resources

Turbonomic monitors the following resources for the cloud target supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz)

	• Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Datacenter	***NOTE:*** *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.* • Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s)

	• Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Kilobytes (KB)
Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)

vCloud Director

A vCloud Director environment has one or more vCloud Director servers (sometimes called cells) to build a multi-tenant private cloud by pooling resources managed by vCenter Servers. For typical installations, each vCloud Director cell manages one vCenter Server instance.

A Provider Virtual Datacenter combines the compute and memory resources from a single vCenter Server with the storage available to that vCenter Server into a single pool of resources to provide to an organization.

An Organization Virtual Datacenter uses the resources in a Provider Virtual Datacenter to present virtual compute and storage to the user.

Prerequisites

- vCloud Director account with System Administrator role
- vCenter service account with privileges to execute Turbonomic actions. For information about service accounts on vCenter Server, see Creating A Service User Account In vCenter on page 25

Adding vCloud Director Targets

> **NOTE:** If any of the vCenter Servers that are connected to vCloud Director are already added as targets in Turbonomic, they must be deleted prior to adding vCloud Director.

A vCloud Director target must specify the address and user of a vCloud Director server and also a service account on the associated vCenter Server instance.

If the credentials for the vCenter Server service account are not valid, then the target will fail to validate in Turbonomic.

To add vCloud Director targets, select the **Cloud Management > vCloud Director** option on the Target Configuration page, and provide the following information:

- Address

  Host name or IP address of the vCloud Director Server.
- Username

  The username for an account on the vCloud Director server that has a System Administrator role.
- Password

  The password for the System Administrator account.
- VC User Name (optional)

  If you provide a value, the username for a service account on the vCenter Server that is managed by this vCloud Director.

  If you do not provide a value, Turbonomic will obtain the username from the vCloud Director server. Even if you do not provide a VC User Name value, you still must provide a VC Password.
- VC Password

  The password for the service account on the vCenter Server.

Actions

Turbonomic recommends actions for the cloud target supply chain as follows.

Entity Type	Action
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Physical Machines	• Start Physical Machine • Provision Physical Machine • Suspend Physical Machine
Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC

Monitored Resources

Turbonomic monitors the following resources for the cloud target supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)

Datacenter

NOTE: *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.*

- Memory (Mem)
  The utilization of the PM's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the PM's CPU reserved or in use
  Measured in Megahertz (MHz)
- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared of memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Kilobytes (KB)

Provider Virtual Datacenter

- Memory (Mem)
  The utilization of the Datacenter's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the Datacenter's CPU reserved or in use
  Measured in Megahertz (MHz)
- Storage
  The utilization of the storage attached to the Provider vDC.
  Measured in Kilobytes (KB)

Consumer Virtual Datacenter

- Memory (Mem)
  The utilization of the Datacenter's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the Datacenter's CPU reserved or in use
  Measured in Megahertz (MHz)
- Storage
  The utilization of the storage attached to the Consumer vDC.
  Measured in Kilobytes (KB)

Virtual Machine Manager

In a VMM environment, the VMM management server processes commands and controls communications with the Hyper-V hosts. To manage VMM, you set the management server as a target. Turbonomic communicates with that target, and also with the Hyper-V hosts that the VMM server manages. For this reason, if you add a VMM target, you should not also add individual Hyper-V hosts as targets. You must grant Turbonomic access to the VMM management server, and also to all the associated Hyper-V machines.

Prerequisites

- Configure remote management on the VMM management Hyper-V server. Refer to the relevant section below:
    - **Turbonomic** 5.4 or higher: See Creating A Service User Account on page 17
    - **Turbonomic** 5.3 or below: See Enabling Management via DCOM on page 189
- Apply necessary hot fixes on the VMM host

  For example, if you are running VMM Server on a Windows Server version earlier than Windows Server 2012 R2, you must apply the hotfix referenced in the Microsoft Knowledge Base article #2842230 (http://support.microsoft.com/kb/2842230).
- Apply time synchronization

  The VM that hosts Turbonomic must be synchronized with each target VMM management server. The Turbonomic Installation Guide includes instructions for synchronizing the clock on the Turbonomic server.
- PowerShell execution must be enabled on the VMM management server.
- Configure port access

  WinRM uses ports 5985 and 5986 for standard and secure communications, respectively. The firewall on your VMM server must open these ports.
- Preexisting Hyper-V Targets

  If any of the hosts that make up the VMM target were added separately as Hyper-V targets (and you do not plan to exclude these hosts), these targets must be deleted. Failure to do so will create duplicate entities in the market, which will negatively impact Turbonomic performance.

Adding VMM Targets

Turbonomic uses the address and credentials you provide to discover the VMM target. From the VMM target, Turbonomic gets the list of managed Hyper-V instances. It then uses that list to discover each Hyper-V instance. The Hyper-V credentials you provide must be valid for all of these machines.

NOTE: In Turbonomic 5.8 and above, Turbonomic will not display the underlying Hyper-V hosts in the Target Configuration page, only the VMM target that was added.

To add VMM targets, select the **Cloud Management > VMM** option on the Target Configuration page, and provide the following information:

- The IP address or host name of the VMM management server
- Which port to use for the WSMan connection

  For a standard connection (HTTP) use 5985. For a secure connection (HTTPS) use 5986.
- Enable or disable a secure connection

  If you enable a secure connection, then you must configure a certificate, and you must configure Turbonomic to communicate over HTTPS. For more information, see Secure Setup of WSMan on page 190.

  Note that setting a secure connection for VMM does not also set secure connections for the underlying Hyper-V hosts. Any communications between Turbonomic and VMM will be secure. To configure secure connections to the underlying Hyper-V hosts, you must specify secure connections on each one.

- Full domain name for the user account

  This domain name identifies the user account for Active Directory authentication.
- Login credentials for the Hyper-V servers that are managed by the VMM target (Optional)

  Turbonomic must log into the Hyper-V servers that the VMM server manages. If you leave the Hyper-V credentials blank, then it will use the same credentials that VMM uses. If you provide Hyper-V credentials, then it will use that service account to log into every Hyper-V managed by the VMM.

  Note that the service account Turbonomic uses to log into a Hyper-V host must satisfy certain requirements. For more information, see "Creating a Service User Account in Hyper-V" in the Turbonomic User Guide.
- Hosts in the VMM target that you would like to exclude from Turbonomic.

  Turbonomic will not discover or manage any hosts you enter in this field. These hosts do not participate in the market, and Turbonomic does not generate actions that involve these hosts. Enter multiple hosts in a comma-separated list. This field accepts both host names, and FQDNs (Fully Qualified Domain Names), and allows the use of the wildcard characters * and ?.

Microsoft VMM and Hyper-V VLANs

Windows Server Hyper-V provides support for VLANs on host and VM partitions. If your Hyper-V environment makes use of this VLAN support, then your VM moves must be sensitive to which hosts provide networking access to your defined VM networks. If a VM is a member of a given VM network, then any move of that VM must be to a host that has access to the same network.

For Hyper-V targets in a VMM environment, the Turbonomic is aware of the VM networks, and ensures that a move is to a host that provides connectivity over the given VM network. For example, if Host 1 and 2 provides connectivity to VM Network A, and Host 3 and 4 provide connectivity to VM Network B, Turbonomic will never recommend a VM with access to Network A, residing on Host 1, to move to Host 3. That would render the VM unable to communicate on Network A.

Configuring SMB 3.0 File Shares Discovery

With VMM, Turbonomic can discover SMB 3.0 shares as datastores, assuming these shares have been properly added to your VMM service center. When you add shares to your VMM environment, be sure to:

- Use the Fully Qualified Domain Name of the file server

  As you the associated file server to your VMM environment (via the Add Storage Devices Wizard), be sure to specify the FQDN of the file server on the Specify Discovery Scope page of the wizard. Do not use the file server's IP address.
- Ensure that file server names are unique

  Do not specify file servers with the same name, even if they belong to different domains. Turbonomic requires the file server names to be unique.

For information about setting up SMB 3.0 shares, please refer to your Microsoft documentation. For example, see "How to Assign SMB 3.0 File Shares to Hyper-V Hosts and Clusters in VMM".

VMM Probe Properties

Turbonomic offers the capability to adjust certain properties of the probe. In general, this tweaking will not be necessary. If adjustment is needed, the file can be found in the `/srv/tomcat/webapps/vmturbo/probe-jars/{probe}/resources` folder. This target has the following properties:

Setting	Function
maxEnvelopeSize	Maximum size of SOAP response envelope
timeout	Determines the timeout in milliseconds for both initial connection and discovery, and maximum period between data packets
executorThreadPoolSize	Size of the thread pool for Discovery process
hypervExecutorThreadPoolSize	Size of the thread pool for host discovery within the VMM target. One thread per cluster, and one per standalone host
vmmExecutorThreadPoolSize	Size of the thread pool used to execute VMM commands in parallel
connectorFactoryThreadPoolSize	Size of the thread pool used to create connection instances
vmmRequestChunkSize	Number of VMM commands issued in one chunk
shellRequestPollerInterval	Time in milliseconds to poll for completion of PowerShell commands
shellRequestTimeout	Time in milliseconds to retrieve result of PowerShell commands
conversionObjectsBatchSize	Maximum amount of internal objects that should be converted into EntityDTO in one batch
queriesThreadPoolCoreSize	Number of threads that will remain in the thread pool even if idle timeout is exceeded. This number should be lower or equal to `queriesThreadPoolMaximumSize`
queriesThreadPoolMaximumSize	Maximum number of threads that will be used by the thread pool
queriesThreadPoolIdleTimeout	Time in seconds before a thread will time out if number of active threads are above `queriesThreadPoolCoreSize`

hostThreadPoolCoreSize	Number of threads that will remain in the thread pool responsible for simultaneous discovery of host instances. This number should be lower or equal to `hostThreadPoolMaximumSize`
hostThreadPoolMaximumSize	Maximum number of threads that will be used by the thread pool responsible for simultaneous discovery of host instances
hostThreadPoolIdleTimeout	Time in seconds before a thread will time out if number of active threads are above `hostThreadPoolCoreSize`
clusterThreadPoolCoreSize	Number of threads that will remain in the thread pool for handling requests to Hyper-V targets. This number should be lower or equal to `clusterThreadPoolMaximumSize`
clusterThreadPoolMaximumSize	Maximum number of threads that will be used by the thread pool responsible for handling requests to Hyper-V hosts
clusterThreadPoolIdleTimeout	Time in seconds before a thread will time out if number of active threads are above `clusterThreadPoolCoreSize`
hypervCommonThreadPoolCoreSize	Number of threads that will remain in the thread pool responsible for miscellaneous SDK tasks
hypervCommonThreadPoolMaximumSize	Maximum number of threads that will be used by the thread pool responsible for miscellaneous SDK tasks. This number should be lower or equal to `hypervCommonThreadPoolMaximumSize`
hypervCommonThreadPoolIdleTimeout	Time in seconds before a thread will time out if number of active threads are above `hypervCommonThreadPoolCoreSize`
actionStepCheckSucceededTimeout	Time in milliseconds to wait before triggering the next step in an action chain
actionStepCheckAmount	Number of times to poll for an action is complete before failure
actionStepCheckTimeout	Time in milliseconds between action checks
requestsPerHost	Maximum number of simultaneous requests per host

Actions

Turbonomic recommends actions for the cloud target supply chain as follows.

Entity Type	Action
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Physical Machines	• Start Physical Machine • Provision Physical Machine • Suspend Physical Machine
Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC

Monitored Resources

Turbonomic monitors the following resources for the cloud target supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s)

- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Megahertz (MHz)

Datacenter

NOTE: *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.*

- Memory (Mem)
  The utilization of the PM's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the PM's CPU reserved or in use
  Measured in Megahertz (MHz)
- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared of memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Kilobytes (KB)

Provider Virtual Datacenter

- Memory (Mem)
  The utilization of the Datacenter's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the Datacenter's CPU reserved or in use
  Measured in Megahertz (MHz)

	• Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)

Public Cloud

Using the Cloud Naive Edition of Turbonomic, public cloud provides compute, storage, and other resources on demand. You can run all of your infrastructure on a public cloud, or you can set up a hybrid environment where you burst workload to the public cloud as needed. Turbonomic can analyze the performance of applications running on the public cloud, and provision more instances as demand requires. For a hybrid environment, Turbonomic can provision copies of your application VMs on the public cloud to satisfy spikes in demand, and as demand falls off it can suspend those VMs if they’re no longer needed.

On the public cloud, you can use Turbonomic to:

- Extend resource allocation across hybrid clouds
- Locate the most efficient workload placement within the hybrid environment, while assuring performance
- Perform elastic load balancing for application groups deployed to the public cloud

Cloud-based datacenters support scalability, resource pooling, multi-tenancy, and self-service management of virtual resources. Turbonomic supports the following cloud technologies:

Supply Chain

For public clouds, Turbonomic discovers Regions and Zones, and adds them as Datacenter Entities. Regions and zones divide the public cloud into managed subsets. A region is typically associated with the geographic location of the cloud resources, and a zone is some division within the region. One region contains multiple zones.

Entity Type	Action
Zone	• Start VM on the Zone • Suspend VM on the Zone

Monitored Resources

Turbonomic monitors the following public cloud infrastructure resources:

Entity Type	Commodity
Zone	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Region	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB)

- CPU Ready
  The utilization of the PM's allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Megahertz (MHz)

Amazon Web Services

Amazon Web Services (AWS) provides a reliable and scalable infrastructure platform in the cloud. You gain access to this infrastructure through a subscription account. To specify an AWS target, you provide the credentials for that account and Turbonomic discovers the resources available to you through that account.

Adding AWS Targets

For Turbonomic to manage an AWS account, you provide the Access Key credentials that you use to access that account. For information about getting an Access Key for an AWS account, see the Amazon Web Services documentation.

To add AWS targets, select **Cloud Management > AWS** on the Target Configuration page, and provide the following information:

- Address
  You can leave this blank, or enter `aws.amazon.com`
- Username
  Provide the **Access Key** for the account you want to manage.
- Password
  Provide the **Access Key Secret** for the account you want to manage.
- Cost and Usage Report Bucket
  Name of the S3 bucket that contains the AWS Cost and Usage report.
- Cost and Usage Report Region
  Region of the S3 bucket that contains the AWS Cost and Usage report.
- Cost and Usage Report Path
  Path in the S3 bucket to the AWS Cost and Usage report.

Cost and Usage Report

In order for Turbonomic to display month-to-day spend, you must create a cost and usage report in AWS and store it in an S3 bucket.

For more information, see Displaying AWS Spend In Turbonomic.

Actions

Turbonomic recommends actions for the cloud target supply chain as follows.

Entity Type	Action
Virtual Machines (Cloud)	• Resize up to template (based on VMem / VCPU) • Resize down to template (based on VMem / VCPU) • Move Virtual Machine (intra-cloud)

NOTE: This is a destructive move- data / applications are not preserved. This action also requires both a Turbonomic merge policy and the moved VM to be a linux VM with template configuration.

Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC
Zone	• Start VM on the Zone • Suspend VM on the Zone

Monitored Resources

Turbonomic monitors the following resources for the cloud target supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Datacenter	***NOTE:*** *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.* • Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz)

	• IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Kilobytes (KB)
Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)
DPod	• Memory (Mem) The utilization of the DPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the DPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the DPod Measured in Kilobytes (KB)

- Flow
  The utilization of the network flow capacity utilized by the DPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization
  Measured in Kilobytes per second (KB/s)

IBM SoftLayer

IBM SoftLayer is an infrastructure platform for the public cloud. You gain access to this infrastructure through a subscription account. To specify a SoftLayer target, you provide the credentials for that account and Turbonomic discovers the resources available to you through that account.

Adding SoftLayer Targets

To add SoftLayer targets, select **Cloud Management > Softlayer** on the Target Configuration page, and provide the following information:

- Address
  You can leave this blank, or enter `control.softlayer.com`
- Username
  For this field, provide the user name for the account you want to manage.
- Password
  Provide the Authentication Key for the account you want to manage.

Actions

Turbonomic recommends actions for the cloud target supply chain as follows.

Entity Type	Action
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC
Zone	• Start VM on the Zone • Suspend VM on the Zone

Monitored Resources

Turbonomic monitors the following resources for the cloud target supply chain:

Entity Type **Commodity**

VirtualMachine

- Virtual Memory (VMem)
  The utilization of the VMem allocated to the hosting VM
  Measured in Kilobytes (KB)
- Virtual CPU (VCPU)
  The utilization of the VCPU allocated to the hosting VM
  Measured in Megahertz (MHz)
- Virtual Storage (VStorage)
  The utilization of the virtual storage capacity allocated for the VM
  Measured in Kilobytes (KB)
- Storage Access Operations Per Second (IOPS)
  The utilization of IOPS allocated for the VStorage on the VM
  Measured in IOPS
- Latency
  The utilization of latency allocated for the VStorage on the VM
  Measured in milliseconds (ms)

Datacenter

NOTE: *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.*

- Memory (Mem)
  The utilization of the PM's memory reserved or in use
  Measured in Kilobytes (KB)
- CPU
  The utilization of the PM's CPU reserved or in use
  Measured in Megahertz (MHz)
- IO
  The utilization of the PM's IO adapters
  Measured in Kilobytes per second (KB/s)
- Net
  The utilization of data through the PM's network adapters
  Measured in Kilobytes per second (KB/s)
- Swap
  The utilization of the PM's swap space
  Measured in Kilobytes (KB)
- Balloon
  The utilization of shared of memory among VMs running on the host. ESX-only
  Measured in Kilobytes (KB)
- CPU Ready
  The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only
  Measured in Kilobytes (KB)

Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)
DPod	• Memory (Mem) The utilization of the DPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the DPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the DPod Measured in Kilobytes (KB) • Flow The utilization of the network flow capacity utilized by the DPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization Measured in Kilobytes per second (KB/s)
Zone	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s)

	• Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Region	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)

Microsoft Azure

Microsoft Azure is Microsoft’s infrastructure platform for the public cloud. You gain access to this infrastructure through a subscription account. To specify an Azure target, you provide the credentials for that account and Turbonomic discovers the resources available to you through that account.

It's common for Cloud operators to host Web applications in Azure. For tips on configuring Web applications to manage with Turbonomic, see the Green Circle article, “How to create and configure a Web Application in Azure for VMTurbo OpsManager 5.6+”.

Prerequisites

- External access via Client/App ID
- Azure Resource Manager ***NOTE:*** *Turbonomic will not discover Azure Classic virtual machines, as they do not utilize the Azure Resource Manager.*

External Access via Client/App ID

The administrator of an Azure account can register a Web App with the account — This gives an external application access to the account's resources. Turbonomic connects to an Azure target via a Web App registered for that Azure account.

To register a Web App with your account:

- Log into the Azure Management Portal
- Add a Web App to an available Active Directory — The Active Directory will correspond with the Tenant ID that you set for the target.
- From the Application's page in the Management Portal, go to the Configure page
  This page displays the generated Client App ID and Client Secret Key for this app.

NOTE: The person who registers the Web App must record the generated secret key. You can return to the Management Portal to see the Client App ID, but the portal only shows the Secret Key when you register that app.

For more information, refer to your Microsoft Azure documentation.

Adding Azure Targets

To add Azure targets, select **Cloud Management > Azure** on the Target Configuration page, and provide the following information:

NOTE: For information about how to get these credentials, see the Green Circle article, "How to get the required Azure credentials to add an Azure Target to VMTurbo OpsManager 5.6+"

- Address
  You can leave this blank, or enter `management.core.windows.net`
- Subscription ID
  The ID for your Azure subscription.
- Tenant ID
  The ID of the tenant you want to manage in your Azure subscription.
- Client App ID
  The Client/App ID of the Web App that gives Turbonomic access to resources in your Azure subscription.
- Client Secret Key
  The secret key for the Web App.

Actions

Turbonomic recommends actions for the cloud target supply chain as follows.

Entity Type	Action
Virtual Machines (Cloud)	• Resize up to template (based on VMem / VCPU) • Resize down to template (based on VMem / VCPU) • Move Virtual Machine (intra-cloud)

NOTE: This is a destructive move- data / applications are not preserved. This action also requires both a Turbonomic merge policy and the moved VM to be a linux VM with template configuration.

Consumer Virtual Datacenters	• Resize Consumer vDC • Provision Consumer vDC

Monitored Resources

Turbonomic monitors the following resources for the cloud target supply chain:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Datacenter	***NOTE:*** *For datacenter entities, Turbonomic does not monitor resources directly from the datacenter, but from the physical machines in the datacenter.* • Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s)

	• Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared of memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Kilobytes (KB)
Provider Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Provider vDC. Measured in Kilobytes (KB)
Consumer Virtual Datacenter	• Memory (Mem) The utilization of the Datacenter's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the Datacenter's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the Consumer vDC. Measured in Kilobytes (KB)
DPod	• Memory (Mem) The utilization of the DPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the DPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the DPod Measured in Kilobytes (KB) • Flow The utilization of the network flow capacity utilized by the DPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization Measured in Kilobytes per second (KB/s)

Storage Targets

Adding a storage Target enables Turbonomic to connect to your storage subsystem through an SMI-S provider API or through the controller's native API. Turbonomic uses the target's API to access information about each of the underlying disk arrays, and uses this information to set disk performance characteristics according to the type and capacity of storage. This leads to improved workload placement. Similarly, Turbonomic knows the relationships between storage controllers and disk arrays, and about the location of datastores within those arrays. This information also helps optimize workload placement.

The section below describes the storage supply chain. For information on how to add specific storage targets, the resources Turbonomic can monitor for the various supply chain entities, and the actions it can take to optimize the environment, refer to the target configuration instructions for your specific storage type.

Supply Chain

Storage targets (storage controllers) add Storage Controller and Disk Array entities to the supply chain. Disk Array entities in turn host Storage entities (datastores).

Entity Mapping

Turbo-nomic	Dell Compellent	EMC VMAX	EMC VNX	EMC XtremIO	HP 3Par	NetApp	Nutanix	Pure
Storage	Volume	Volume (Regular, Thin, Meta)	LUN, Data Store, Volume	Volume	Virtual Volume	Volume	Container	Volume
Disk Array	Storage Type	Disk Group or Thin Pool	Raid Group, Storage Pool	XTremIO Cluster	CPG	Aggregate	Storage Pool	Shelf Array
Storage Controller	Storage Center	VMAX Array	Storage Processor	XTremIO Cluster	Controller	Controller / Filer	Controller VM	Controller

Actions

Turbonomic recommends actions for storage targets as follows.

NOTE: This is a general list of actions for storage managed by storage controllers. Specific actions Turbonomic can recommend, and which actions it can automate depends on the actual technology — Not all actions make sense for all types of storage. For example, Turbonomic can automate a datastore move across disk arrays or storage controllers for NetApp in C mode, but not for other storage technologies. For another example, Turbonomic doesn't recommend an action to provision a datastore on Nutanix because that platform includes datastores with each node — to increase storage capacity you would provision a Nutanix node.

You can see how actions differ per technology in each section that describes adding a specific type of Storage Manager target.

Entity Type	Action
Storage	• Start Storage • Provision Storage • Suspend Storage • Move (only with Storage Targets configured) • Resize (only with Storage Targets configured)
Disk Arrays	• Provision Disk Array • Start Disk Array • Suspend Disk Array • Move Disk Array (for NetApp Cluster-Mode, only) • Move Virtual Machine • Move Datastore
Storage Controller	• Provision Storage Controller (recommendation only)

Monitored Resources

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• CPU The utilization of the Storage Controller's CPU in use Measured in Megahertz (MHz) • Storage Amount The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller Measured in Megabytes (MB)

Dell Compellent

Turbonomic supports the management of Dell SC Series (Compellent) disk arrays and storage controllers. Turbonomic connects through the Dell Enterprise Manager and performs management as a client of the Enterprise Manager Data Collector.

The Dell Enterprise Manager is a management service that provides administration, management, and monitoring of multiple Storage Centers — Typically installed on a Windows VM.

When you specify a Dell Compellent target, you provide the IP address of the Dell Enterprise Manager. Turbonomic discovers the Compellent infrastructure through the SMI-S component which is typically installed as part of the Enterprise Manager.

NOTE: Before adding the Dell Compellent target to Turbonomic, confirm that the Storage Centers you want to manage show up in Dell Enterprise Manager (see “Storage Center Administration” in the *Dell Compellent Enterprise Manager Administrator's Guide*). The SMI-S user account must be able to access all of the Storage Centers. If you add or remove Storage Centers later, Turbonomic will detect the changes during its next discovery cycle.

Prerequisites

- Dell Enterprise Manager Data Collector Service 6.2 or higher
- Dell Compellent SMI-S Provider
- Storage Centers added to Dell Enterprise Manager

Setting Up the Dell Compellent SMI-S Provider

Your Dell Compellent storage environment must include an enabled Dell Compellent SMI-S Provider. Configure the SMI-S Provider as described in the “SMI-S” section of the *Dell Compellent Enterprise Manager Administrator's Guide*. The guide provides detailed steps to:

- Open the required ports on the server hosting the Enterprise Manager Data Collector.
- Enable SMI-S for the Data Collector.
- Add a user for SMI-S.
- If using HTTPS, associate the SSL certificate with the SMI-S Provider.

For more information, some customers find the following Green Circle article useful: Configuring VMTurbo Operations Manager to discover Dell Compellent Targets

Adding Dell Compellent Targets

To add Dell Compellent targets, select the **Storage > Dell Compellent** option on the Target Configuration page and provide the following information:

- Address
  The name or IP address of the Dell Enterprise Manager.
  By default, Enterprise Manager provides SMI-S data over port 5988 (HTTP) or port 5989 (HTTPS). If your installation uses a different port for SMI-S, include the port number in the Address field.
- Username/Password
  Credentials for the SMI-S user you added when setting up the SMI-S provider.

After validating the new target, Turbonomic discovers the connected storage entities. This table compares terms used in the Dell Enterprise Manager to those used in Turbonomic:

Dell Name	Turbonomic Entity
Storage Center	Storage Controller
Storage Type	Disk Array
Volume	Storage

Supply Chain

Storage targets (storage controllers) add Storage Controller and Disk Array entities to the supply chain. Disk Array entities then host Storage entities (datastores). For a visual representation, see the introductory Supply Chain on page 66.

Supported Actions

Turbonomic supports the following actions for Dell Compellent entities:

Entity Type	Can Be Automated	Recommendations only
Storage		Move, Provision, Resize Up
Disk Array		Provision, Resize Up
Storage Controller		Provision

Monitored Resources

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• CPU The utilization of the Storage Controller's CPU in use Measured in Megahertz (MHz) • Storage Amount The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller Measured in Megabytes (MB)

VMAX

Turbonomic supports management of VMAX2 and 3 Series storage arrays. The VMAX series is a family of enterprise storage arrays designed for SAN environments. Turbonomic connects to VMAX storage systems via an EMC SMI-S provider that has the disk arrays added to it. A single SMI-S provider can communicate with one or more disk arrays. When you specify an SMI-S provider as a target, Turbonomic discovers all the added disk arrays.

Turbonomic will create Storage Groups based on the SLO levels defined in VMAX3 Targets. By default, Storage Move actions will respect these SLO levels based on the configured response time.

Prerequisites

- EMC SMI-S Provider 8.1
- A service account that Turbonomic can use to connect to the EMC SMI-S Provider (typically the default `admin` account)

Adding VMAX Targets

To add VMAX targets, select the **Storage > VMAX** option on the Target Configuration page and provide the following information:

- Address
  The IP or hostname of the SMI-S provider. If using HTTPS, you must follow the IP with `:PORT`.
- Username
  The Username for the SMI-S provider.
- Password
  The Password for the SMI-S provider.

Entity Comparison

After validating the new target, Turbonomic discovers the connected storage entities. This table compares terms used in EMC VMAX to those used in Turbonomic:

EMC VMAX Name	Turbonomic Entity
Volume (Regular, Thin, Meta)	Storage
Storage Resource Pool (VMAX3) / Thick Provisioned Pool (earlier)	Disk Array
Storage Group (VMAX3) / Thin Provisioned Pool (earlier)	Logical Pool
VMAX Array	Storage Controller

Supported Actions

For each discovered entity, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations only
Storage	Provision (Cloning), Delete	Resize (V-Volumes only)
Logical Pool		Resize
Disk Array		Provision

Monitored Resources

When calculating available storage, Turbonomic excludes disks devoted to the VMAX operating system by default. If these disks are assigned to new raid groups or storage pools, the capacity of those disks will then be considered when calculating the capacity of the Storage Controller.

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Logical Pool	• Storage Amount The utilization of the logical pool's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the logical pool's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the logical pool. Measured in Operations per second • Latency The utilization of latency on the logical pool. Measured in milliseconds (ms)

Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• Storage Amount The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller Measured in Megabytes (MB)

EMC VNX

Turbonomic supports management of EMC VNX (not VNXe) file and block level storage systems, as well as unified VNX configurations. The requirements for specifying a VNX target differ, depending on the type of storage system you want to manage:

- File level storage
  Turbonomic connects to the VNX Control Station to discover and collect data from connected storage volumes.
- Block level storage
  Turbonomic uses the EMC SMI-S Provider to discover and collect data from block-level disk arrays and the controllers/storage processors in the VNX environment. A single SMI-S provider can communicate with multiple arrays.
- Unified configuration
  For unified (block and file) implementations, configure one VNX target for block and another VNX target for file.

Prerequisites

- For block level storage or unified configurations: EMC SMI-S Provider 4.6.2 (See below)
- For file level storage, VNX Control Station
- Firewall ports for the SMI-S provider
  On the system that hosts the SMI-S provider you may need to open the following ports:
    - UDP port 427
    - TCP ports 5988 and 5989

- A service account Turbonomic can use to connect to the target:
    - File level storage: An account with administrator privileges on the VNX Control Station (`nasadmin`).
    - Block level storage: An account with administrator privileges on the SMI-S provider host.

EMC SMI-S Provider Setup Requirements

NOTE: The SMI-S Provider is required for block level storage and unified configurations only. If you are using file level storage only, you do not need the SMI-S Provider.

1. **Download EMC SMI-S Provider 4.6.2 from the EMC Support website.**
2. **Install the SMI-S Provider.**

   Perform installation according to the installation instructions in the EMC SMI-S Provider Release Notes. When installing the provider:
    - In the Providers list, select the “Array Provider” (not the “Host Provider”).
    - If prompted to select which Stordaemon Services to install and start, select “EMC SE SYMAPI Server Daemon.”
3. **Add storage resources to the SMI-S provider.**

   Use the “testsmiprovider” utility to add the disk arrays/pools you want to manage. Find instructions in the “Out-of-band discovery method” section of the EMC SMI-S Provider Release Notes.

The SMI-S Provider communicates over UDP port 427 and TCP ports 5988 and 5989. If a firewall is running on the server hosting the provider, these ports must be open. Also, the SMI-S Provider service must be running at all times.

NOTE: If you are running a Vblock environment or are using the Cisco UCS Converged Fabric, you may be using a Cisco SMI-S provider. Turbonomic requires the EMC SMI-S Provider. Since both providers use the same TCP port, install the EMC provider on its own system.

Adding EMC VNX Targets

To add EMC VNX targets, select the **Storage > EMC VNX** option on the Target Configuration page and provide the following information:

- Address

  File level storage: The address of the primary VNX Control Station. You can get this from the EMC Unisphere management console.

  Block level storage: The name or IP address of the server with the EMC SMI-S provider (see the section above for SMI-S Provider Setup Requirements).
- Username/Password

  File level storage: Credentials for an administrator user on the primary VNX Control Station.

  Block level storage: The name or IP address of the EMC SMI-S provider (see the section above for SMI-S Provider Setup Requirements).

NOTE: For unified configurations, configure one VNX target for block and another VNX target for file.

After validating the new target, Turbonomic discovers the connected storage entities. This table compares terms used in EMC VNX to those used in Turbonomic:

VNX Name (CLARiiON / Celerra)	Turbonomic Entity
LUN, Data Store / Volume	Storage
Raid Group, Storage Pool / Storage Pool	Disk Array
Storage Processor / Storage Processor	Storage Controller

Supply Chain

Storage targets (storage controllers) add Storage Controller and Disk Array entities to the supply chain. Disk Array entities then host Storage entities (datastores). For a visual representation, see the introductory Supply Chain on page 66.

Supported Actions

For each discovered entity, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations only
Storage		Move, Provision, Resize Up
Disk Array		Provision, Resize Up
Storage Controller		Provision

Monitored Resources

When calculating available storage, Turbonomic excludes disks devoted to the VNX operating system by default. If these disks are assigned to new raid groups or storage pools, the capacity of those disks will then be considered when calculating the capacity of the Storage Controller.

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second

	• Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• CPU The utilization of the Storage Controller's CPU in use Measured in Megahertz (MHz) • Storage Amount The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller Measured in Megabytes (MB)

EMC VPLEX

Turbonomic supports management of EMC VPLEX virtual storage systems in a local configuration, via the VPLEX API. Currently, Turbonomic does not support Metro or Geo configurations.

VPLEX is used to aggregate and refine data collected between connected Storage and Hypervisor targets. VPLEX supports one-to-one, one-to-many, and many-to-one relationships between virtual volumes and LUNs. Only one-to-one mapping between virtual volume and LUNs is supported by Turbonomic.

Prerequisites

- VPLEX Management Server
- Hypervisor target supported by Turbonomic
- Storage target supported by Turbonomic

NOTE: In order for Turbonomic to make use of the information provided by VPLEX, you must also add the hypervisor and storage layered under it as targets.

Adding EMC VPLEX Targets

To add EMC VPLEX targets, select the **Storage > EMC VPLEX** option on the Target Configuration page and provide the following information:

- Address:
  The IP or Hostname of the VPLEX Management Server
- Username:
  The Username for the VPLEX Management Server
- Password:
  The Password for the VPLEX Management Server
- Port Number:
  The port number for the remote management connection. The default port number for the VPLEX Management server is 443.

Supported Actions

For this target, actions are generated and executed via the underlying storage targets. Turbonomic will use the enhanced visibility provided by VPLEX to make more intelligent storage decisions- for example, recommending storage vMotion between pools.

EMC XtremIO

EMC® XtremIO® is a flash-based (SSD) storage solution, designed to push data to applications at higher speeds. The system building blocks are SAN appliances called X-Bricks. A deployment is organized into clusters of X-Bricks, and the clusters are managed by the XtremIO Management Server (XMS).

Turbonomic connects to X-Bricks through the XMS. The XMS presents a unified view of each connected X-Brick cluster, rather than exposing the individual X-Bricks within each cluster. Within Turbonomic, each X-Brick cluster displays as a single storage controller with an associated disk array.

The relationship between Storage entities and individual X-Bricks within the cluster is not exposed through the XMS — Turbonomic cannot make recommendations to move datastores from one X-Brick to another. Additionally, the X-Brick has a fixed form factor — Turbonomic does not recommend resize actions for disk array or storage controller resources.

Turbonomic recognizes XtremIO arrays as flash storage and sets the IOPS capacity on discovered arrays accordingly.

Prerequisites

- A service user account on XMS 4.0 or higher — typically the default `xmsadmin` account
  Turbonomic uses this account to connect to the XMS and execute commands through the XtremIO API.

Adding XtremIO Targets

For EMC XtremIO targets, select the **Storage > EMC XtremIO** option on the Target Configuration page and provide the following information:

- Address
  The name or IP address of the XtremIO Management Server (XMS).
- Username/Password
  Credentials for a user account on the XMS.

After validating the new target, Turbonomic discovers the connected storage entities. This table compares terms used in XTremIO to those used in Turbonomic:

XTremIO Name	Turbonomic Entity
Volume	Storage
XTremIO Cluster	Disk Array
XTremIO Cluster	Storage Controller

Supply Chain

Storage targets (storage controllers) add Storage Controller and Disk Array entities to the supply chain. Disk Array entities then host Storage entities (datastores). For a visual representation, see the introductory Supply Chain on page 66.

Supported Actions

For each discovered entity, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations only
Storage		Provision, Resize Up
Disk Array		
Storage Controller		Provision

Monitored Resources

When calculating available storage, Turbonomic excludes disks devoted to the VNX operating system.

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB)

	• Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• CPU The utilization of the Storage Controller's CPU in use Measured in Megahertz (MHz) • Storage Amount The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller Measured in Megabytes (MB)

HP 3PAR

HP 3PAR StoreServ systems use controller nodes to manage pools of storage resources and present a single storage system to consumers. Turbonomic communicates with the HP 3PAR system via an SMI-S provider that is installed on the 3PAR controller node.

Prerequisites

- SMI-S Provider enabled and configured on the controller node
- A service account on the controller node that Turbonomic can use to connect to the SMI-S provider

Setting Up the SMI-S Provider

The HP 3PAR SMI-S Provider should be installed on the controller node. It is disabled by default — you must ensure that it is installed properly and running on the controller node.

To enable the SMI-S provider:

1. **Log into the HP 3PAR Command Line Interface (CLI).**
   Open a secure shell session (ssh) on the controller node. Default credentials are `3paradm/3pardata`.
2. **Check the current status of the SMI-S provider.**
   In the shell session, execute the command, `showcim`.
3. **If the CIM service is not running, start it.**
   Execute the command `startcim` to enable the CIM service and the SMI-S provider.

To stop the SMI-S provider, execute the command `stopcim -f -x`.

Adding HP 3PAR Targets

To add an HP 3PAR target, select the **Storage > HP 3Par** option on the Target Configuration page and provide the following information:

- Address
  The name or IP address of the 3PAR controller node.
  By default, the controller provides SMI-S data over port 5988 (HTTP) or port 5989 (HTTPS). If your installation uses a different port for SMI-S, include the port number in the Address field.
- Username/Password
  Credentials for a user account on the controller node.

After validating the new target, Turbonomic discovers the connected storage entities. This table compares terms used in HP 3PAR to those used in Turbonomic:

HP 3PAR Name	Turbonomic Entity
Virtual Volume	Storage
CPG	Disk Array
Controller	Storage Controller

Supply Chain

Storage targets (storage controllers) add Storage Controller and Disk Array entities to the supply chain. Disk Array entities then host Storage entities (datastores). For a visual representation, see the introductory Supply Chain on page 66.

3Par Adaptive Optimization

Adaptive Optimization (AO) for HP 3Par enables management of data storage across two or three tiers. AO places storage regions on the appropriate tier in response to periodic analysis that AO performs.

To work with the storage in an AO group, Turbonomic:

- Discovers each Common Provisioning Group (CPG) as a disk array
  In the Turbonomic user interface, these disk arrays do not host storage — They appear empty. Turbonomic will not recommend storage moves between these disk arrays, because such moves would conflict with AO block-level placement.
- Creates a single AO disk array that hosts all the datastores in an AO group
  This disk array represents the AO group, and it includes all the member CPGs. Turbonomic considers this single disk array when it performs analysis — It can recommend moving storage into or out of the AO group. Also, Turbonomic aggregates resource capacity in this disk array. For example, the IOPS capacity for the AO disk array is a combination of IOPS capacity for the constituent CPGs.

You can see the AO disk array in the Turbonomic user interface. The display name for this disk array:

- Begins with the string `AO_DiskArray`
- Includes the names of the discovered disk arrays (CPGs), separated by underscores

Supported Actions

For each discovered entity, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations only
Storage		Move, Provision, Resize Up
Disk Array		Resize Up
Storage Controller		Provision

Monitored Resources

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second

	• Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• CPU The utilization of the Storage Controller's CPU in use Measured in Megahertz (MHz) • Storage Amount The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller Measured in Megabytes (MB)

Hitachi Data Systems

Turbonomic supports management of Hitachi Data Systems (HDS) storage systems, via either the embedded SMI-S, or the shared SMI-S (Not recommended). If using the embedded SMI-S provider, each Storage System must be added to Turbonomic as a separate target.

Prerequisites

- SMI-S Provider
  Turbonomic recommends using the embeded SMI-S provider. Turbonomic also provides the ability to use the shared SMI-S Provider. To use the shared SMI-S provider, please contact Turbonomic support. For information about using SMI-S with HDS storage, see The Hitachi Command Suite Administrator Guide.
- Tuning Manager
  The Tuning Manager is required for monitoring IOPs and Latency. For assistance in setting up the Tuning Manager, see The HDS Tuning Manager User Guide.

Adding HDS Targets

To add HDS targets, select the **Storage > HDS** option on the Target Configuration page and provide the following information:

- SMI-S Address
  The IP or hostname of the SMI-S provider.
- SMI-S Username
  The Username for the SMI-S provider.
- SMI-S Password
  The Password for the SMI-S provider.
- SMI-S Secure connection
  If you are using HTTPS with the SMI-S Provider, select this.
- Shared SMI-S Provider
  If you are using the Shared SMI-S Provider (not recommended), select this.
- TnM Address
  The IP or hostname of the Tuning Manager.
- TnM Username
  The Username for the Tuning Manager.
- TnM Password
  The Password for the Tuning Manager.
- TnM Secure connection
  If you are using HTTPS with the Tuning Manager, select this.

Entity Comparison

After validating the new target, Turbonomic discovers the connected storage entities. This table compares terms used in HDS to those used in Turbonomic:

HDS Name	Turbonomic Entity
Volume, Logical Device (LDEV), V-VOL	Storage
Parity Group	Disk Array
Dynamic Provisioning Pool	Logical Pool
Storage System	Storage Controller

Supported Actions

For each discovered entity, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations only
Storage		Provision, Suspend, Resize (V-Volumes only), Move
Logical Pool		Provision, Resize, Delete
Disk Array		Provision
Storage Controller		

Monitored Resources

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Logical Pool	• Storage Amount The utilization of the logical pool's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the logical pool's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the logical pool. Measured in Operations per second • Latency The utilization of latency on the logical pool. Measured in milliseconds (ms)

Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• CPU The utilization of the Storage Controller's CPU in use Measured in Megahertz (MHz) • Storage Amount The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller Measured in Megabytes (MB)

NetApp

Storage control adds support for NetApp filers running the Data ONTAP operating system. NetApp storage controllers are Storage Virtual Machines that manage storage arrays — Vfilers for 7-Mode and Vservers for C-Mode. Turbonomic connects to these storage controllers to support NetApp targets in 7-Mode and Cluster-Mode (C-Mode).

Prerequisites

- Transport Layer Security (TLS) is enabled
- A service account Turbonomic can use to connect to the NetApp target

Enabling TLS

Starting with version 5.4, by default Turbonomic requires Transport Layer Security (TLS) version 1.2 to establish secure communications with targets. NetApp filers have TLS disabled by default, and the latest version they support is TLSv1. If your NetApp target fails to validate on Turbonomic 5.4 or later, this is probably the cause.

If target validation fails because of TLS support, you might see validation errors with the following strings:

- No appropriate protocol
  To correct this error, ensure that you have enabled the latest version of TLS that your target technology supports. If this does not resolve the issue, please contact Turbonomic Technical Support.
- Certificates does not conform to algorithm constraints
  To correct this error, refer to your NetApp documentation for instructions to generate a certification key with a length of 1024 or greater on your target server. If this does not resolve the issue, please contact Turbonomic Technical Support.

For information about enabling TLS, see the Data ONTAP **System Administration Guide** for sections on the SSL protocol.

Service User Account — Administrator Role

To discover and fully manage NetApp disk arrays, Turbonomic must have a service account that grants privileges to execute commands through the NetApp filer’s OnTap API (ontapi). In most cases, you can provide a user account with Administrator privileges:

- **NetApp 7-Mode**: Create the administrator account from the NetApp command line — For example:
  ```
  useradmin user add Turbonomic -g Administrators
  ```
- **NetApp C-Mode**: Create the administrator account via the NetApp OnCommand System Manager, or from the NetApp command line — For example:
  ```
  security login create -role admin -username Turbonomic -application ontapi -
  authmethod password
  ```

If you prefer not to grant full administrator rights, see Restricted Service Accounts In NetApp on page 89

Adding NetApp Targets

To add a NetApp target, select the **Storage > NetApp** option on the Target Configuration page and provide the following information:

- Address
  The name or IP address of the NetApp Storage Controller.
  7-Mode: Enter the storage controller address.
  Cluster-Mode (C-Mode): Enter the cluster management address.
- Username/Password
  Credentials for the NetApp service user account that you have configured for Turbonomic to use.

After validating the new target, Turbonomic discovers the connected storage entities. This table compares terms used in NetApp to those used in Turbonomic:

NetApp Name	Turbonomic Entity
Volume	Storage
Aggregate	Disk Array
Controller / Filer	Storage Controller

Supply Chain

Storage targets (storage controllers) add Storage Controller and Disk Array entities to the supply chain. Disk Array entities then host Storage entities (datastores). For a visual representation, see the introductory Supply Chain on page 66.

Supported Actions

For each discovered entity, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations only
Storage	Move (C-Mode only)	Move (7-Mode), Provision, Resize Up
Disk Array		Resize Up, Move (C-Mode only), Provision (C-Mode only)
Storage Controller		Provision

Note that for NetApp in C-Mode, Turbonomic can automate moving a datastore to a disk array on the same storage controller, as well as moves to a disk array on a different storage controller.

Monitored Resources

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB)

	• Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• CPU The utilization of the Storage Controller's CPU in use Measured in Megahertz (MHz) • Storage Amount The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller Measured in Megabytes (MB)

Restricted Service Accounts In NetApp

While Turbonomic prefers a NetApp service account with administrator rights, it is possible to create an account that has limited access, by following the steps outlined below, depending on NetApp mode.

NetApp 7-Mode Restricted Service Account Setup

If you prefer to use a service account that does not have full administrator rights:

1. **Log into the NetApp filer from a command shell.**
2. **Create a role with API privileges.**

   For example:

   ```
   useradmin role add Turbonomic Role <capabilities>]
   ```

   where `<capabilities>` is a comma-separated list of capabilities assigned to the role. The required capabilities are listed below.

   Inspection capabilities:

   - `api-aggr-list-info`
   - `api-disk-list-info`
   - `api-fcp-node-get-name`
   - `api-flash-device-list-info`
   - `api-igroup-list-info`
   - `api-iscsi-node-get-name`
   - `api-lun-initiator-list-map-info`
   - `api-lun-map-list-info`
   - `api-lun-list-info`
   - `api-net-ifconfig-get`
   - `api-nfs-exportfs-list-rules-2`
   - `api-options-list-info`
   - `api-system-get-info`
   - `api-system-get-version`
   - `api-volume-list-info`

- `api-snapshot-list-info`
- `api-perf-object-get-instances`
- `api-perf-object-instance-list-info`
- `api-perf-object-counter-list-info`
- `api-qtree-list`
- `security-api-vfiler`
- `api-vfiler-list-info`
- `api-volume-options-list-info`
- `login-http-admin`
- `login-*`

Note that the last login capability (`login-*`) may be necessary for external users.

Execution capabilities:

- `api-volume-create`
- `api-volume-size`
- `api-volume-offline`
- `api-volume-online`
- `api-volume-destroy`
- `api-aggr-add`
- `api-aggr-create`
- `api-aggr-offline`
- `api-aggr-online`
- `api-aggr-destroy`

3. **Create a group and assign the role.**

   For example:

   ```
   useradmin add Turbonomic Group -r Turbonomic Role
   ```

4. **Create a user that is a member of the group.**

   For example:

   ```
   useradmin user add Turbonomic -g Turbonomic Group
   ```

5. **Enter a password for the new user when prompted.**

NetApp C-Mode Restricted Service Account Setup

If you prefer to use a service account that does not have full administrator rights:

1. **Log into the NetApp filer from a command shell.**
2. **Create a role and assign it permission to execute each of the following commands:**
   - `aggr-get-iter`
   - `igroup-get-iter`
   - `cluster-identity-get`
   - `lun-map-get-iter`
   - `net-interface-get-iter`
   - `storage-disk-get-iter`
   - `system-get-node-info-iter`
   - `volume-get-iter`
   - `vserver-get-iter`
   - `fcp-node-get-name`
   - `flash-device-get-iter`

- `iscsi-node-get-name`
- `options-list-info`
- `qtree-list-iter`
- `system-get-version`
- `lun-get-iter`
- `snapshot-get-iter`
- `perf-object-get-instances`
- `volume-get-iter`
- `volume-move-get-iter`
- `volume-move-start`

For example, to enable volume offline, execute the following:

```
security login role create -role Turbonomic Role -access all -cmddirname
"volume offline" -vserver <cluster_name>
```

3. **Create a user based on the role you create.**

Give the user access to the `ssh` and `ontapi` applications. For example:

```
security login create -role Turbonomic Role -username Turbonomic -application ontapi
-authmethod password
```

```
security login create -role VMTurboRole -username VMTurbo -application ssh -authmethod
password
```

Nutanix

Nutanix products provide hyperconverged platforms that include VM hosting and a distributed storage fabric. The platform presents storage in two tiers — Local HDD storage and server-attached flash (hot storage).

Nutanix environments include:

- One or more Nutanix appliances

  An appliance contains up to four server nodes.
- Nutanix nodes

  Servers that expose compute and storage resources. Each node provides local HDD and hot storage. Nodes combine to form a unified cluster that pools resources.
- Controller VMs

  Each node includes a Controller VM that manages the node’s resources within the cluster pool. To minimize storage latency, the Controller VM keeps the most frequently accessed data in the hot storage.

Turbonomic supports management of Nutanix fabrics, where the supply chain treats a Nutanix Storage Pool as a disk array. Turbonomic recognizes Nutanix storage tiers when calculating placement of VMs and VStorage. In addition, Turbonomic can recommend actions to scale flash capacity up or down by adding more hosts to the cluster, or more flash drives to the hosts.

To specify a Nutanix target, provide the Cluster External IP address. This is a logical IP address that always connects to one of the active Controller VMs in the cluster. In this way, you can specify a Nutanix target without having to specify an explicit Controller VM.

NOTE: The Controller VM must remain *pinned* to its host machine — You must not move the Controller VM to a different host. If the Nutanix cluster uses the Nutanix Acropolis OS to manage VMs, Turbonomic automatically pins the Controller VMs. However, if you use vCenter Server or Hyper-V to manage VMs on the hosts, you must configure a group to pin the Controller VMs. For more information, see Pinning Nutanix Controller VMs in Generic Hypervisor Mode on page 96.

Prerequisites

- A service account with administrator rights on the Nutanix target

Finding the Cluster External IP Address

To set a Nutanix target, provide the Cluster External IP address for the given Nutanix cluster.

The Cluster External IP address is a logical IP that resolves to the cluster's Prism Leader. If the Prism Leader fails, then the Cluster External IP address will resolve to the newly elected Prism Leader.

To find this IP address, open the Web Console (the Prism Element) on the cluster and navigate to the **Cluster Details** view. In this view you can see the **Cluster External IP** address. If there is no IP address specified, you can specify the address at this time. For more information, see the Nutanix documentation.

Operating Modes

A Nutanix node is a server that hosts VMs — In this sense the node functions as a hypervisor. A cluster of nodes can host VMs using the following Hypervisor technologies:

- Nutanix Acropolis
  The native Nutanix host platform, which combines software-defined storage with built-in virtualization.
- VMware ESXi
- Microsoft Hyper-V

Turbonomic divides Nutanix cluster management into two modes:

- Standalone Mode (Acropolis)
  In this mode you:
    - Enable standalone mode and restart the Turbonomic server
    - Specify the Nutanix Cluster External IP address as the target address — This adds the cluster as a Turbonomic target to manage both VM and storage resources
- Generic Hypervisor Mode (ESXi or Hyper-V)
  In this mode you:
    - Add each Hyper-V host or vCenter as a hypervisor target — This enables VM workload control for the respective hypervisor technologies
    - Specify the Nutanix Cluster External IP address as the target address — This adds the cluster as a Storage Controller target to enable Turbonomic storage control

NOTE: These operating modes are mutually exclusive for a Nutanix cluster. You cannot mix operating modes in the same cluster.

Enabling Standalone (Acropolis) Operating Mode

By default, Turbonomic supports the Generic Hypervisor operating mode. To enable management of Acropolis VMs, you must modify the Turbonomic targets configuration file and then restart the Turbonomic server, as outlined in Enabling Standalone (Acropolis) Operating Mode In Nutanix on page 95.

Controller VM Pinning

Each Nutanix node hosts a Controller VM that runs the Nutanix software and manages I/O for the hypervisor and all VMs running on the host. Each Controller VM must remain on its host node —The Controller VM must be *pinned* to that host, and must not be moved to any other host.

For more information about how to pin the Controller VM, see Pinning Nutanix Controller VMs in Generic Hypervisor Mode on page 96.

Adding Nutanix Targets

NOTE: This describes how to add a Nutanix cluster to Turbonomic as a target. The steps are the same no matter which operating mode you use (Standalone or Generic Hypervisor). Before you add the cluster as a target, you should know which operating mode you intend. If you want Standalone mode, then you will have to enable that operating mode after adding the cluster. If you want Generic Hypervisor mode, then you will have to add the hypervisors as targets after you have added the Nutanix cluster as a target. For more information, see Hypervisor Targets on page 5.

To add Nutanix targets, select the **Storage > Nutanix** option on the Target Configuration page and provide the following information:

- Address
  The Cluster External IP address for the Nutanix cluster.
- Username/Password
  Credentials for an account on the Nutanix cluster.

After validating the new target, Turbonomic discovers the connected storage entities. This table compares terms used in Nutanix to those used in Turbonomic:

Nutanix Name	Turbonomic Entity
Container	Storage
Storage Pool	Disk Array
Controller VM	Storage Controller

Supply Chain

Storage targets (storage controllers) add Storage Controller and Disk Array entities to the supply chain. Disk Array entities then host Storage entities (datastores). For a visual representation, see the introductory Supply Chain on page 66.

Supported Actions

For each discovered entity, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations only
Datastore ("Storage")		Provision, Resize Up
Disk Array		
Storage Controller		Provision

Monitored Resources

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• CPU The utilization of the Storage Controller's CPU in use Measured in Megahertz (MHz)

- Storage Amount
  The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller

  Measured in Megabytes (MB)

Enabling Standalone (Acropolis) Operating Mode In Nutanix

> **NOTE:** Standalone and Generic modes are mutually exclusive for a Nutanix cluster. You cannot mix operating modes in the same cluster.

To enable Acropolis management for a cluster:

1. **Add the Nutanix cluster as a target.**

   Add the cluster as a target using the Cluster External IP address. Be sure you have added the target, and that Turbonomic has validated the target. Keep a note of the IP address you used.

2. **Open a secure session on the Turbonomic server.**

   ```
   ssh root@<server_ip_address>
   ```

   The default password is `vmturbo`.

3. **Navigate to the configuration directory.**

   Change directory to `/srv/tomcat/data/config`.

4. **Back up the config file.**

   Create a backup of `disc.config.topology`.

5. **Locate the target entry for the cluster you added.**

   Open `disc.config.topology` in a text editor.

   In this file, search the `<targets>` section for the `nutanix:NutanixTarget` entry that has the IP address you supplied for the Nutanix cluster.

   For example, assume you added a Nutinax cluster with the IP address, `10.10.64.88`. Then you would search for the following entry in the config file:

   ```
   <targets xsi:type="nutanix:NutanixTarget" uuid="_pRSdsFheEeWgpfmffIwmXw"
   name="VMTTarget_10.10.64.88" displayName="10.10.64.88" nameOrAddress="10.10.64.88"
   timeout="30000" template="_2kieMIMcEeCC5PhGZd6dJj" ... </targets>
   ```

6. **Modify the entry’s TEMPLATE attribute.**

   Note the template attribute. In the example above, the value is `_2kieMIMcEeCC5PhGZd6dJj`.

   Change the last character in the `template` attribute from `j` to `i`. Following the example, the config file entry would now be:

   ```
   <targets xsi:type="nutanix:NutanixTarget" uuid="_pRSdsFheEeWgpfmffIwmXw"
   name="VMTTarget_10.10.64.88" displayName="10.10.64.88"
   nameOrAddress=“10.10.64.88" timeout="30000"
   template="_2kieMIMcEeCC5PhGZd6dJi" ... </targets>
   ```

7. **Save your changes to the configuration file.**

8. **Restart Turbonomic.**

Pinning Nutanix Controller VMs in Generic Hypervisor Mode

Each Nutanix node hosts a Controller VM that runs the Nutanix software and manages I/O for the hypervisor and all VMs running on the host. Each Controller VM must remain on its host node —The Controller VM must be *pinned* to that host, and must not be moved to any other host.

For a cluster in Standalone mode (running Acropolis hypervisors), Turbonomic recognizes the Controller VMs, and never recommends move actions for them.

For a cluster in Generic Hypervisor mode (using vCenter or Hyper-V hypervisors), you must use Turbonomic policies to pin the Controller VMs to their respective nodes. To do this, you will create a dynamic group of Nutanix Controller VMs, and then disable move actions for all members of this group.

To pin the Controller VMs:

1. **Create a group of Controller VMs.**

   In Turbonomic you can create dynamic groups based on VM name — All VMs with matching names automatically belong to the group. Nutanix uses the following naming convention for Control VMs: `NTNX-<SerialNumber>-A-CVM`, where `<SerialNumber>` is the serial number of the Controller VM.

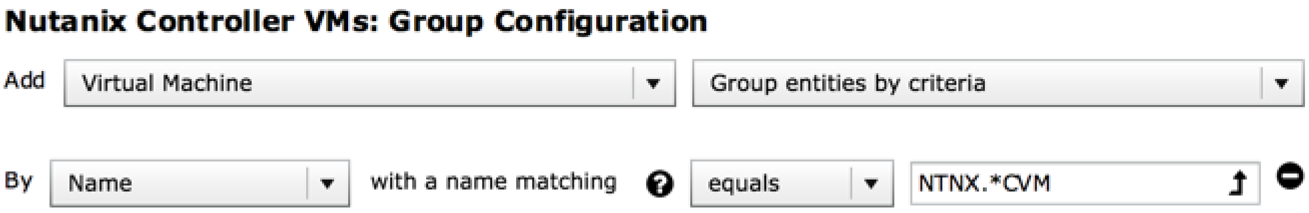
   - Create a new group

     In Turbonomic go to the **Policy > Group Management** view and create a new group that groups VM entities by criteria.

   - Add a filter to match the VM names

     Add a filter that matches names by the regular expression, `NTNX.*CVM`. This regular expression will match the Nutanix Controller VMs.

   The group specification should be similar to the following:



Nutanix Controller VMs: Group Configuration

Add: Virtual Machine — Group entities by criteria

By: Name — with a name matching — equals — NTNX.*CVM

   Be sure to save the group. All the Nutanix Controller VMs will automatically become members of this group.

2. **Disable moves for all VMs in this group.**

   - In Turbonomic go to the Policy > Action > VM view
   - Set the scope to the group you made

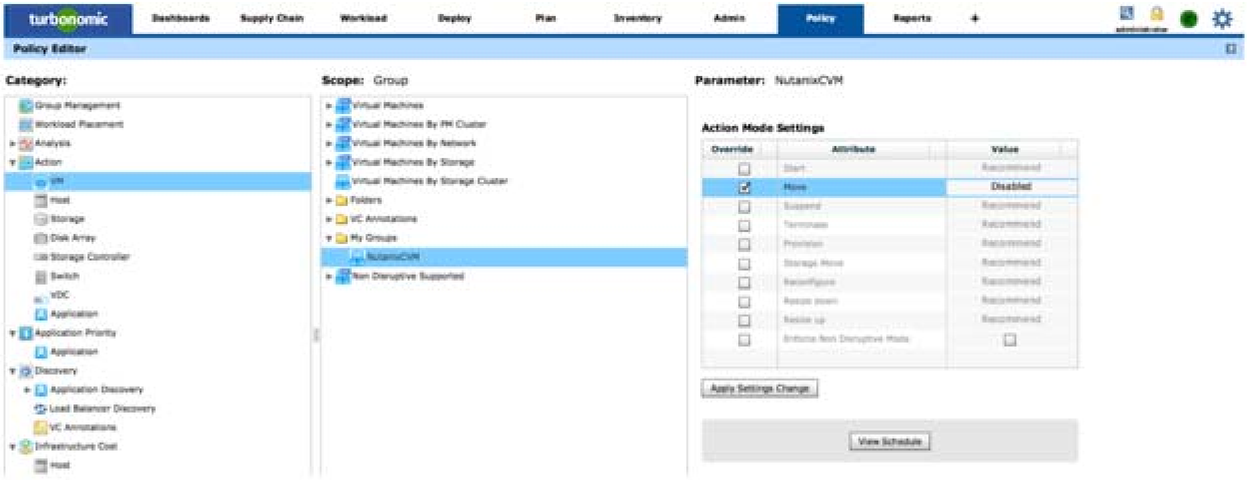
     In the **Scope** column, expand **My Groups** and select the group you just made.

   - Disable moves for this group

     In the **Parameter** column under **Action Mode Settings**, set the value to **Disabled**. This will override the global action mode.

   - Save the action mode settings

     Be sure to click **Apply Settings Change**.



Scope: Group

Parameter: NutanixCVM

Action Mode Settings

Override	Attribute	Value
☐	Start	Recommend
☑	Move	Disabled
☐	Suspend	Recommend
☐	Terminate	Recommend
☐	Provision	Recommend
☐	Storage Move	Recommend
☐	Reconfigure	Recommend
☐	Resize down	Recommend
☐	Resize up	Recommend
☐	Enforce Non Disruptive Mode	☐

Pure Storage

Turbonomic supports management of Pure Storage FlashArray systems. Note that one Pure Storage target manages a single Pure Storage FlashArray instance. The storage devices in the array are all flash storage — to analyze IOPS capacity, Turbonomic uses the setting made for SSD Disks in the Policy View.

Because of the improved performance of Pure Storage arrays, Turbonomic intelligently moves more demanding workloads to these datastores. Turbonomic analysis is also able to incorporate Pure Storage de-duplication and compression when recommending actions.

Prerequisites

- A service account Turbonomic can use to connect to the FlashArray
  This account needs privileges to execute commands through the Pure Storage API — Typically the default `pureuser` administrative account.

Adding Pure Storage Targets

To add a Pure Storage target, select the **Storage > Pure Storage** option on the Target Configuration page and provide the following information:

- Address
  The name or IP address of the Pure Storage FlashArray.
- Username/Password
  Credentials for the service account Turbonomic can use to connect to the FlashArray.

After validating the new target, Turbonomic discovers the connected storage entities. This table compares terms used in Pure to those used in Turbonomic:

Pure Name	Turbonomic Entity
Volume	Storage
Shelf Array	Disk Array
Controller	Storage Controller

Supply Chain

Storage targets (storage controllers) add Storage Controller and Disk Array entities to the supply chain. Disk Array entities then host Storage entities (datastores). For a visual representation, see the introductory Supply Chain on page 66.

Supported Actions

For each discovered entity, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Can Be Automated	Recommendations only
Storage		Resize Up
Disk Array		
Storage Controller		Provision

Pure Storage assigns all the disks managed by a storage controller to a single array, with a fixed form-factor. There are no actions to perform for an array — For example, there is no action to move a disk array from one storage controller to another. Likewise, there are no actions to move or provision volumes because of the fixed form-factor.

Monitored Resources

Turbonomic monitors the following storage resources:

Entity Type	Commodity
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second

	• Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
Disk Array	• Storage Amount The utilization of the Disk Array's capacity. Measured in Megabytes (MB) • Storage Provisioned The utilization of the Disk Array's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the disk array Measured in Operations per second • Latency The utilization of latency on the disk array. Measured in milliseconds (ms)
Storage Controller	• CPU The utilization of the Storage Controller's CPU in use Measured in Megahertz (MHz) • Storage Amount The utilization of the storage controller's capacity. The storage allocated to a storage controller is the total of all the physical space available to aggregates managed by that storage controller Measured in Megabytes (MB)

Guest OS Process Targets

This version of Turbonomic supports the following Guest OS Process targets:

- SNMP (Simple Network Management Protocol)
- WMI (Windows Management Instrumentation)
- Docker

NOTE: Starting with version 5.8, Turbonomic offers Target Configuration for WMI, SNMP, and Docker targets. This method deprecates discovery via application policies. To avoid duplication of any legacy targets, you must remove them from the Policy view before adding them via Target Configuration.

Do not discover the same targets though both Policy view and Target Configuration. Doing so will cause entity duplication and may adversely affect your environment.

Supply Chain

Guest Process targets add Application entities to the supply chain. You can navigate to each individual application server to see specific details.

Actions

Turbonomic recommends actions for the Guest OS Process supply chain as follows.

NOTE: This is a general list of actions for entities discovered for applications. You can see how actions differ per technology in each section that describes adding a specific type of application target.

Entity Type	Action
Applications	Without discovered Guest OS Processes, Turbonomic doesn't perform actions on applications. Instead, it performs resize actions on the host VMs. If host utilization is high enough on the physical machine running the application VM, Turbonomic may also recommend provisioning a new host. For specific applications, see the individual application entry.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the Guest OS Process supply chain:

Entity Type	Commodity
WMI/SNMP Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Docker Containers

Turbonomic will discover Docker application containers running on VMs in your entire environment, or a portion of your environment, based on scope.

Prerequisites for Docker Containers

- Underlying VM host targets added to Turbonomic (for discovery)
- Docker remote API (Docker REST service) v1.20+ enabled/configured on target VMs

---

NOTE: For Hyper-V hosts, you must install Hyper-V Integration Services on the target VMs. For more information, please refer to the following integration services TechNet article:

`https://technet.microsoft.com/en-us/library/dn798297%28v=ws.11%29.aspx`

Adding Docker Targets

To add Docker targets, select the **Guest OS Processes > Docker** option on the Target Configuration page and provide the following information:

- Target ID
  The display name that will be used to identify the target in the Target List. This is for display in the UI only; it does not need to match any internal name.
- Port
  The Port number Turbonomic will use to connect to the Virtual Machine hosting the application
- Secure Connection
  Turbonomic will use SSL to connect to the target
- Scope
  Turbonomic will search for Docker containers on the VMs found within the set scope. This scope can be set to the entire environment, single or multiple clusters, or particular physical machines. Select this option and choose the scope for discovery.

Supported Actions

For each discovered entity within the container supply chain, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Action
Docker Container	• Resize Container • Recommend Only
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the container supply chain:

Entity Type	Commodity
Docker Container	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

SNMP

Turbonomic will discover application and operating system resources using SNMP (Simple Network Management Protocol) on Linux hosts in your entire environment, or a portion of your environment, based on scope.

Prerequisites

- Underlying VM host targets added to Turbonomic (for discovery)
- SNMP service enabled/configured on target VMs

NOTE: For Hyper-V hosts, you must install Hyper-V Integration Services on the target VMs. For more information, please refer to the following integration services TechNet article:

`https://technet.microsoft.com/en-us/library/dn798297%28v=ws.11%29.aspx`

For VMware hosts, you must install VMware Tools on the target VMs.

Adding SNMP Targets

To add SNMP targets, select the **Guest OS Processes > SNMP** option on the Target Configuration page and provide the following information:

- Target ID
  The display name that will be used to identify the target in the Target List. This is for display in the UI only; it does not need to match any internal name.
- Community
  The SNMP v2c community string Turbonomic will use to connect to the application
- Port number
  The Port number Turbonomic will use to connect to the Virtual Machine hosting the application
- Scope
  Turbonomic will search for SNMP applications using the port entered above on the VMs found within the set scope. This scope can be set to the entire environment, single or multiple clusters, or particular physical machines. Select this option and choose the scope for application discovery.

  Turbonomic recommends that a single scope contain no more than 500 virtual machines. Multiple targets should be used for environments exceeding this limit.

SNMP Probe Properties

Turbonomic offers the capability to adjust certain properties of the probe. In general, this tweaking will not be necessary. If adjustment is needed, the file can be found in the `/srv/tomcat/webapps/vmturbo/probe-jars/{probe}/resources` folder. This target has the following properties:

Setting	Function
connectorTimeoutMillis	Maximum time to wait for a response from a single request
snmpConnectorRetries	Number of attempts the SNMP connector will make to connect

Supported Actions

For each discovered entity within the application supply chain, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Action
Applications	Without discovered Guest OS Processes, Turbonomic doesn’t perform actions on applications. Instead, it performs resize actions on the host VMs. If host utilization is high enough on the physical machine running the application VM, Turbonomic may also recommend provisioning a new host. For specific applications, see the individual application entry.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application supply chain:

Entity Type	Commodity
WMI/SNMP Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS

- Latency
  The utilization of latency allocated for the VStorage on the VM
  Measured in milliseconds (ms)

Enabling SNMP

Turbonomic requires that SNMP is enabled and configured in order to discover Guest Processes. While these steps will change slightly between OS versions, the general instructions are below:

1. **Obtain and install the SNMP server package for your version of Linux. This is commonly called** `net-snmp` in most package managers. Also install the `net-snmp-utils` **package if it is available**
2. **Configure the SNMP deamon by editing** `/etc/snmp/snmpd.conf` or running `snmpconf -i` **at the command line in order to verify that:**
   - SNMPv2c is enabled
   - A read-only community name has been set. This community name will be used by Turbonomic to communicate with the VM

NOTE: The community name you select must have at least read-only access to all OIDs in the system.

3. **Configure the daemon to listen on a public interface. Most default installations only listen on** `127.0.0.1`
4. **Start the SNMP daemon process**

Sample snmp.conf Files

Provided below are two sample `snmp.conf` files that will aid in making the necessary changes.

Long form, VACM:

```
# First, map the community name "turbonomic" into a "security name"
# sec.name   source         community
com2sec notConfigUser  default      turbonomic
# Second, map the security name into a group name:
#       groupName      securityModel securityName
group   notConfigGroup v2c           notConfigUser
# Finally, grant the group read-only access to any OID.
#       group          context sec.model sec.level prefix read   write  notif
access  notConfigGroup ""      any       noauth    exact  all none none
```

Short form, older syntax:

```
rocommunity turbonomic default system
```

Verify SNMP Setup

Verify that your SNMP setup is successful by using the following command from a remote machine, replacing the community string and IP address:

```
snmpwalk -Os -v 2c -c COMMUNITY_STRING IP_ADDRESS iso.3.6.1.2.1.1.1
```

If successful, the command will return the kernel version of the machine (similar to the output of `uname -a`)

WMI

Turbonomic will discover application and operating system resources using WMI (Windows Management Instrumentation) in your entire environment, or a portion of your environment, based on scope.

Prerequisites

- Underlying VM host targets added to Turbonomic (for discovery)
- WMI Enabled on target VMs. For assistance in enabling WMI, see "Enabling WMI" in the Turbonomic User Guide.

NOTE: For Hyper-V hosts, you must install Hyper-V Integration Services on the target VMs. For more information, please refer to the following integration services TechNet article:

`https://technet.microsoft.com/en-us/library/dn798297%28v=ws.11%29.aspx`

For VMware hosts, you must install VMware Tools on the target VMs.

Adding WMI Targets

To add WMI targets, select the **Guest OS Processes > WMI** option on the Target Configuration page and provide the following information:

- Target ID

  The display name that will be used to identify the target in the Target List. This is for display in the UI only; it does not need to match any internal name.
- Username

  The username Turbonomic will use to connect to the Virtual Machine hosting the application
- Password

  The password Turbonomic will use to connect to the Virtual Machine hosting the application
- Secure Connection

  When selected, Turbonomic will connect using SSL
- Domain Name:

  The Active Directory domain used by Turbonomic in conjuction with the Username for authentication. Leave blank for local accounts.
- Scope:

  Turbonomic will search for WMI applications on the VMs found within the set scope. This scope can be set to the entire environment, single or multiple clusters, or particular physical machines. Select this option and choose the scope for application discovery.

  Turbonomic recommends that a single scope contain no more than 500 virtual machines. Multiple targets should be used for environments exceeding this limit.

WMI Probe Properties

Turbonomic offers the capability to adjust certain properties of the probe. In general, this tweaking will not be necessary. If adjustment is needed, the file can be found in the `/srv/tomcat/webapps/vmturbo/probe-jars/{probe}/resources` folder. This target has the following properties:

Setting	Function
connectorTimeoutMillis	Maximum time to wait for a response from a single request
slaCapacity	Capacity for the SLA commodity
transactionCapacity	Capacity for the Transaction commodity

Supported Actions

For each discovered entity within the application supply chain, Turbonomic can execute or recommend certain actions, as outlined below.

Entity Type	Action
Applications	Without discovered Guest OS Processes, Turbonomic doesn’t perform actions on applications. Instead, it performs resize actions on the host VMs. If host utilization is high enough on the physical machine running the application VM, Turbonomic may also recommend provisioning a new host. For specific applications, see the individual application entry.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the container supply chain:

Entity Type	Commodity
WMI/SNMP Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB)

	• Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Enabling WMI

Turbonomic requires that WMI is enabled and configured in order to discover Windows-based Guest Processes. While these steps will change slightly between Windows Server versions, the general instructions are below:

1. **Enable WMI by navigating to** `Computer Management > Services and Applications > WMI Control`**.**
2. **Enable the following services, and set them to start automatically: Remote Registry, Windows Management Instrumentation, and Windows Remote Management.**
3. **Allow Jinterop access through the firewall.**
4. **Execute Regedit and search for the following:** `76a64158-cb41-11d1-8b02-00600806d9b6`**.**
5. **Change the owner and permissions on each occurance found by this search to give the user ownership and full control.**
6. **Restart the Remote Registry service.**

After restarting the Remote Registry service, the WMI targets will be available for discovery.

Application Server Targets

This version of Turbonomic supports the following application servers as targets:

- IBM WebSphere
- Oracle WebLogic
- JBoss
- Apache Tomcat
- Generic JVM

With the exception of Apache Tomcat and Generic JVM, these target types support domains of application servers that are controlled by management servers. For such managed domains you can add the management server as a target, and Turbonomic will discover the managed application servers. You can also add an individual application server as a target, or you can add all matching targets within a given scope.

To add any of these servers as a target, you specify:

- The IP address of the VM hosting the management server
  You can optionally specify the IP address of an individual application server.
- The port that the target server listens on
- Administrator credentials for the target server instance

Supply Chain

Application Server targets add Application entities to the supply chain. The inventory groups applications by server type — You can navigate to each individual application server to see specific details.

Resource Monitoring

To manage application servers, Turbonomic monitors the resources that affect application performance, as listed in the following table. Note that because WebLogic performs its own tuning of thread pools, Turbonomic does not monitor threads for WebLogic application servers.

Monitored Resource	WebSphere	WebLogic	JBoss	Tomcat	JVM
Heap	✓	✓	✓	✓	✓
Garbage Collection Time	✓	✓	✓	✓	✓
Threads	✓	✓	✓	✓	
Transactions	✓	✓	✓	✓	
Response Time	✓	✓	✓	✓	
QoS (a measure of application latency)	✓	✓	✓	✓	
VMem (for underlying VM)	✓	✓	✓	✓	
VCPU (for underlying VM)	✓	✓	✓	✓	

In addition to monitoring these resources and recommending associated actions, Turbonomic:

- Monitors garbage collection in the application memory space, and uses that information when recommending actions for heap resizes.
- Discovers JDBC connection pools and through these pools can discover relationships between an application server and a given database server. As a result, Turbonomic sees the JDBC connection pool as a consumer of database connections, and can recommend resize actions for that connection pool.

Application Server Scaling Policy

After you specify an application server target, Turbonomic discovers the resource utilization for that server, and recommends appropriate actions. These actions will follow one of the two following scaling policies:

- Provision

  Scaling by Provision enables horizontal scaling, where the environment adjusts to increased demand by provisioning new application servers.
- Resize

  Scaling by Resize enables vertical scaling, where the environment adjusts to increased demand by resizing the application server, or the VM that hosts the application server.

These scaling policies are mutually exclusive — the affected application servers can scale by Provisioning or by Resizing, but not both. For example, Turbonomic can recommend resize actions, but if you have set the Scaling Policy to Provision, Turbonomic will only recommend Provision and Suspend actions.

You set the Scaling Policy in the Policy view — Turbonomic policies support scope, so you can select specific groups of application servers to set their Scaling Policies as you like. For more information, see "Application Server Actions" in the Turbonomic User Guide.

Actions

Turbonomic recommends actions for the application supply chain as follows.

NOTE: This is a general list of actions for entities discovered for applications. See specific actions for each target on that targets page.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
Applications	Without discovered Guest OS Processes, Turbonomic doesn’t perform actions on applications. Instead, it performs resize actions on the host VMs. If host utilization is high enough on the physical machine running the application VM, Turbonomic may also recommend provisioning a new host. For specific applications, see the individual application entry.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

NOTE: This is a general list of commodities for entities discovered for application servers. You can see how actions differ per technology in each section that describes adding a specific type of application server target.

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server's heap Measured in Kilobytes (KB) • Response Time The utilization of the server's allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server's thread capacity Measured in Threads • DBMem The utilization of the database's memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server's capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)

VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

JBoss

Turbonomic supports connecting to JBoss targets running in these operation modes:

- Managed Domain
  A collection of JBoss servers in a domain, with a single Domain Controller process acting as the central management control point. In this case, the VM that hosts the Domain Controller will be the target. You configure a Domain Controller via the `domain-controller` entry in the `host.xml` file on the JBoss machine.
- Standalone
  A single JBoss server — The VM that hosts the server is the target.

Prerequisites

- The target VM is properly configured as Domain Controller or standalone, depending on the operations mode
- Discovered infrastructure
  Turbonomic discovers JBoss servers that are running on VMs or containers. The hosting VM or container must already be in your Turbonomic inventory.

  To set the target for a server running on a VM, you must have first discovered the hosting VM through a hypervisor target. To set the target for a server running in a container, you must have configured container discovery for JBoss applications.
    - For information about container targets, see Application Container Targets on page 159
    - For information about hypervisor targets, see Hypervisor Targets on page 5
- To monitor Threads (utilization of thread pool capacity), threads must be configured for the JBoss server
  Turbonomic monitors the Threads resource in application servers to track utilization of thread pool capacity. To monitor threads in JBoss, each JBoss server must define a thread pool in its configuration files. To specify thread pools, see your JBoss documentation.

Adding a JBoss Target

You can add an individual JBoss server as a target, or you can add all matching targets within a given scope.

To add a server as a target, specify:

- **Address:** The name or IP address of the VM hosting the JBoss server
  If you are adding a Domain Controller, give the IP address of the VM that hosts the Domain Controller. For a standalone JBoss server, give the address of the VM that hosts the JBoss server.
- **Scope (optional):** A cluster or group of VMs that host JBoss servers or Domain Controllers
  If you set target scope, Turbonomic scans each VM within that group or cluster and tries to connect to a JBoss server over the specified port. Turbonomic adds any JBoss servers it finds as targets.
- **Port Number:** The port that connects to the JBoss server
  The default port for HTTP access is 9990.
- **Username/Password:** Valid administrator credentials for the JBoss server or Domain Controller

Application Names

Turbonomic displays discovered JBoss servers in the user interface. These names indicate whether they're standalone or in a managed domain.

The name is divided into three sections:

- IP Address
- Domain Controller name or “STANDALONE”
- The server name

Actions

Turbonomic recommends actions for the application supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.

JBoss

- Resize Heap
  This action can only be executed by Turbonomic when running in a domain controller. Standalone applications will see only a recommendation.
- Resize Thread Pool
  Recommendation only.
- Resize Connection Capacity
  Recommendation only.
- Suspend VM
  This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation.
- Provision VM
  This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation.

Virtual Machines

- Provision additional resources (VMem, VCPU)
- Move Virtual Machine
- Move Virtual Machine Storage
- Reconfigure Storage
- Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

Entity Type / Commodity

Virtual Applications

- Transactions
  The utilization of the allocated transactions per second for the given virtual application

  Measured in transactions per second

Application

- Virtual Memory (VMem)
  The utilization of the VMem allocated to the hosting VM

  Measured in Kilobytes (KB)
- Virtual CPU (VCPU)
  The utilization of the VCPU allocated to the hosting VM

  Measured in Megahertz (MHz)
- Transactions
  The utilization of the allocated transactions per second for the given virtual application

  Measured in transactions per second
- Heap
  The utilization of the application server’s heap

  Measured in Kilobytes (KB)
- Response Time
  The utilization of the server’s allocated response time

  Measured in Milliseconds (ms)

	• Threads The utilization of the server’s thread capacity Measured in Threads • DBMem The utilization of the database’s memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server’s capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Apache Tomcat

Turbonomic supports connecting to individual Tomcat targets. Turbonomic connects to the Tomcat process as a remote client via remote JMX access. Target configuration includes the port used by the JMX/RMI registry.

Prerequisites

- A valid service user account on the Tomcat server
  If Tomcat security is enabled, this must be a Tomcat JMX user with a `readonly` role.
- Tomcat should run on JDK version 7 or 8

- For VMware environments, VMware Tools must be installed on the VM that hosts the Tomcat
  This ensures that the VM hosting the Tomcat can get its IP address
- Remote JMX access is enabled through a port that is opened to the firewall
- Discovered infrastructure
  Turbonomic discovers Tomcat servers that are running on VMs or containers. The hosting VM or container must already be in your Turbonomic inventory.

  To set the target for a server running on a VM, you must have first discovered the hosting VM through a Hypervisor target. To set the target for a server running in a container, you must have configured container discovery for Tomcat applications.
    - For information about container targets, see Application Container Targets on page 159
    - For information about hypervisor targets, see Hypervisor Targets on page 5

Configuring JMX Remote Access

Turbonomic monitors and controls the Tomcat server via JMX Remote access. You must configure a JMX Remote port.

Note that to work with a firewall you should also set the RMI Server port — If you don't set an RMI port, then JMX sets an arbitrary *ephemeral port*, and you can't guarantee that the port will be open to your firewall.

There are two ways to set JMX Remote port on Linux platforms:

- Ports specified as system properties
  You can set the port via the system property, `com.sun.management.jmxremote.port`. For example:
  ```
  com.sun.management.jmxremote.port=8050
  ```
  A common way to set this property is to declare it in the `CATALINA_OPTS` system variable — You can set this in the `setenv.sh` script. For example:
  ```
  CATALINA_OPTS="$CATALINA_OPTS
  -Dcom.sun.management.jmxremote
  -Dcom.sun.management.jmxremote.port=8050"
  export CATALINA_OPTS
  ```
  Note that this sets the JMX Remote port, but it does not set the RMI Server port — Tomcat startup will specify an ephemeral port for the RMI server.
- Ports specifed in a JMX Remote Lifecycle Listener
  This listener component fixes the ports used by the JMX/RMI Server. When you configure the listener, you specify both the JMX Remote port and the RMI Server port. This is the preferred method when working with a firewall. For more information, see the Apache Tomcat documentation.

On Windows, the typical installation is with Tomcat as a service. There are two ways to set the JMX Remote port:

- Via setenv.bat
  Add the property to the `CATALINA_OPTS` environment variable:
  ```
  set "CATALINA_OPTS=%CATALINA_OPTS% -Dcom.sun.management.jmxremote.port=8050"
  ```
- Use the Tomcat configuration utility (tomcat7w or tomcat8w)
  Set the port with the following command:
  ```
  -Dcom.sun.management.jmxremote.port=8050"
  ```

To discover the JMX port that is set to an already running Tomcat, you can look in the following locations:

- For Linux platforms, look in the configuration files — Either:
  - `setenv.sh` — Assuming you configured the port by adding it to the `CATALINA_OPTS` environment variable
  - `$CATALINA_HOME/conf/server.xml` — Assuming you configured a JMX Remote Lifecycle Listener in this file
- For Windows platforms, look in:
  - `setenv.bat` — Assuming you configured the port by adding it to the `CATALINA_OPTS` environment variable
  - The Windows registry — Assuming you installed Tomcat as a Windows service using the Tomcat Configuration utility

Adding a Tomcat Target

You can add an individual Tomcat server as a target, or you can add all matching servers within a given scope.

To add a server as a target, specify:

- **Target Name:** Name displayed in the Turbonomic UI
- **Username:** Username of an account with the Admin role
- **Password:** Password of an account with the Admin role
- **Scope:** A cluster or group of VMs that host Tomcat servers
  If you set target scope, Turbonomic scans each VM within that group or cluster and tries to connect to a Tomcat server over the specified port. Turbonomic adds any Tomcat servers it finds as targets.
- **Port Number:** The JMX Remote port

Actions

Turbonomic recommends actions for the application supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.

Tomcat	• Resize Heap Recommendation only. • Resize Thread Pool Recommendation only. • Resize Connection Capacity Recommendation only. • Suspend VM This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation. • Provision VM This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server's heap Measured in Kilobytes (KB) • Response Time The utilization of the server's allocated response time Measured in Milliseconds (ms)

	• Threads The utilization of the server's thread capacity Measured in Threads • DBMem The utilization of the database's memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server's capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

JVM Application

Turbonomic supports connecting to individual JVM Applications as targets. Turbonomic connects to the JVM process as a remote client via remote JMX access. Target configuration includes the port used by the JMX/RMI registry.

Prerequisites

- A valid user account on the JVM application
  If JMX security is enabled this must be a JMX user with a `readonly` role
- The application should run on JDK version 7 or 8

- For VMware environments, VMware Tools must be installed on the VM that hosts the application
  This ensures that the VM hosting the application can get the application's IP address
- Remote JMX access is enabled through a port that is opened to the firewall
- Discovered infrastructure
  Turbonomic discovers JVM applications that are running on VMs or containers. The hosting VM or container must already be in your Turbonomic inventory.

  To set the target for a server running on a VM, you must have first discovered the hosting VM through a Hypervisor target. To set the target for a server running in a container, you must have configured container discovery for JVM applications.
    - For information about container targets, see Application Container Targets on page 159
    - For information about hypervisor targets, see Hypervisor Targets on page 5

Configuring JMX Remote Access

Turbonomic monitors and controls the JVM application via JMX Remote access. You must configure a JMX Remote port.

Note that to work with a firewall you should also set the RMI Server port — If you don't set an RMI port, then JMX sets an arbitrary *ephemeral port*, and you can't guarantee that the port will be open to your firewall.

To set the JMX Remote port, pass in the port at the command line when you start your application. For example, to set the port to `8090`, start your application with the following options:

```
-Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.port=8090
```

Adding a JVM Target

You can add an individual JVM application as a target, or you can add all matching applications within a given scope.

To add an application as a target, specify:

- **Address:** The name or IP address of the VM hosting the JVM application
- **Scope (optional):** a Turbonomic group of VMs that host JVM applications
  If you set target scope, Turbonomic scans each VM within that group or cluster and tries to connect to a JVM application over the specified port. Turbonomic adds any applications it finds as targets.
- **Port Number:** The JMX Remote port
- **Username/Password:** Credentials for a user account with an Admin role
  The credentials you provide must match the credentials you specify for JMX login configuration when you start up the application.

  If you disable authentication on the application, then you must still provide arbitrary values for **Username** and **Password**. To disable JMX authentication, use the following flags in the command line as you start the application:

  ```
  -Dcom.sun.management.jmxremote.authenticate=false
  -Dcom.sun.management.jmxremote.ssl=false
  ```

Multiple JVM Targets On Single VM

Note that you can specify targets with different ports, but that run on the same VM (use the same IP address). You can also specify targets via the same scope, but with different ports — This is another way to assign applications running on the same VM to different ports. To do this:

To do this, add the targets in two separate steps. For example, assume you want to add two JVM application targets, and they both run on the VM at `10.10.123.45`. One application is on port `123`, and the other application is on port `456`. To specify these two targets:

- Specify the first target with the following parameters:
    - Scope/Address: `10.10.123.45`
    - Port number: `123`
    - Username: `AppUser`
    - Password: `**********`

  Then click **ADD** to create an entry in the Pending Targets list.
- Specify the second target with the following parameters:
    - Scope/Address: `10.10.123.45`
    - Port number: `456`
    - Username: `AppUser`
    - Password: `**********`

  Then click **ADD** to create an entry in the Pending Targets list.
- Click **Apply** to add these pending targets to Turbonomic and begin validation and discovery

Actions

Turbonomic recommends actions for the application supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
JVM	• Resize Heap Recommendation only. • Resize Thread Pool Recommendation only. • Resize Connection Capacity Recommendation only.

	• Suspend VM This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation. • Provision VM This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

Entity Type	**Commodity**
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
JVM Application	• Heap The utilization of the application server's heap Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Oracle WebLogic

The typical WebLogic deployment is a managed domain with one Administration Server that provides a single point of entry for administration and management of the domain. The domain can include other WebLogic Servers which are the Managed Servers. You set the WebLogic Administration Server as a Turbonomic target.

For a standalone WebLogic deployment, the single server acts as its own Administration Server — You can set the standalone server as a Turbonomic target.

WebLogic deployments can include clusters to distribute workload across multiple WebLogic servers. Turbonomic recommended actions respect the cluster architecture. For example, if you have enabled horizontal scaling for your WebLogic servers, then Turbonomic can recommend provisioning new servers for a given cluster.

Prerequisites

- A service user account with an Admin role.
  To execute actions the service account must have an Admin role. For read-only monitoring and analysis, you can set the target with a more restricted role, but then you will have to execute all recommended actions manually, through the WebLogic interface.
- Weblogic requires both the local and remote ends of the connection be resolvable by DNS. The Turbonomic IP address must be resolvable by your local DNS server.
- Discovered infrastructure
  Turbonomic discovers WebLogic servers that are running on VMs or containers. The hosting VM or container must already be in your Turbonomic inventory.

  To set the target for a server running on a VM, you must have first discovered the hosting VM through a Hypervisor target. To set the target for a server running in a container, you must have configured container discovery for WebLogic applications.
    - For information about container targets, see Application Container Targets on page 159
    - For information about hypervisor targets, see Hypervisor Targets on page 5

Finding the T3 Listen Port

To configure a WebLogic target, you need to know the port that the server listens on for administrative communications. Launch the WebLogic Administration Console:

- Navigate to Domain Structure and display the domain you’re interested in
- Navigate to **Environment > Servers** and select the Domain Administration Server you’re setting as a target
  The console displays configuration information for the server, including the T3 listen port.

Adding a WebLogic Target

You can add an individual WebLogic server as a target, or you can add all matching targets within a given scope.

To add a server as a target, specify:

- **Address:** The name or IP address of the VM hosting the WebLogic server
- **Scope (optional):** A cluster or group of VMs that host WebLogic servers
  If you set target scope, Turbonomic scans each VM within that group or cluster and tries to connect to a WebLogic server over the specified port. Turbonomic adds any WebLogic servers it finds as targets.
- **Port Number:** The T3 listen port that’s configured on the WebLogic server
  The default port for HTTP access is 9990.
- **Username/Password:** Credentials for a wluser account with an Admin role

Actions

Turbonomic recommends actions for the application supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
WebLogic	• Resize Heap This action can only be executed by Turbonomic when running in a domain controller. Standalone applications will see only a recommendation. • Resize Thread Pool Recommendation only. • Resize Connection Capacity Recommendation only. • Suspend VM This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation. • Provision VM This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server’s heap Measured in Kilobytes (KB) • Response Time The utilization of the server’s allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server’s thread capacity Measured in Threads • DBMem The utilization of the database’s memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server’s capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)

VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

IBM WebSphere

The typical WebSphere deployment is a cell of WebSphere servers, controlled by a Deployment Manager. A cell makes up a managed domain that incorporates multiple VMS that host managed application servers. The Deployment Manager is a WebSphere instance that provides a single point of entry for the managed domain.

To configure the WebShpere installation, you can use the WebShpere Integrated Solutions Console. This is a client that exposes configuration settings including the SOAP port and the PMI settings.

To manage the servers in an installation, WebSphere uses the Performance Monitoring Infrastructure (PMI). Each WebSphere server runs a PMI service that collects performance data from the various application server components. Turbonomic uses PMI for monitoring and control of the WebSphere installation.

Prerequisites

- The PMI service set to monitor at the Basic level or greater
- A service user account with an Administrator role

  To execute actions the service account must have an Administrator role. For read-only monitoring and analysis, you can set the target with a more restricted role (Monitor), but then you will have to execute all recommended actions manually, through the WebSphere interface.
- Discovered infrastructure

  Turbonomic discovers WebSphere servers that are running on VMs or containers. The hosting VM or container must already be in your Turbonomic inventory.

  To set the target for a server running on a VM, you must have first discovered the hosting VM through a Hypervisor target. To set the target for a server running in a container, you must have configured container discovery for WebSphere applications.
    - For information about hypervisor targets, see Application Container Targets on page 159
    - For information about container targets, see Hypervisor Targets on page 5

Finding the SOAP Connector Address

To configure a WebSphere target, you need to know the port that the server listens on for administrative communications. Launch the WebSphere Administration Console:

- Navigate to System **Administration > Deployment Manager**
- Under **Additional Properties**, click **Ports**
  The entry for `SOAP_CONNECTOR_ADDRESS` gives the currently set port number.

Adding a WebSphere Target

You can add an individual WebLogic server as a target, or you can add all matching targets within a given scope.

To add a server as a target, specify:

- **Target Name:** Name displayed in the Turbonomic UI
- **Username:** Username of a wsuser account with the Admin role
- **Password:** Password of a wsuser account with the Admin role
- **Scope:** A cluster or group of VMs that host WebSphere servers
  Turbonomic will scan each VM within that group or cluster and try to connect to a WebSphere server over the specified port. Turbonomic adds any WebSphere servers it finds as targets.
- **Port Number:** Port used by the `SOAP_CONNECTOR_ADDRESS`

Actions

Turbonomic recommends actions for the application supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
WebSphere	• Resize Heap This action can only be executed by Turbonomic when running in a domain controller. Standalone applications will see only a recommendation. • Resize Thread Pool Recommendation only.

	• Resize Connection Capacity Recommendation only. • Suspend VM This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation. • Provision VM This action can only be executed by a VM hosted in a vCenter environment. Applications running on other hypervisors will see only a recommendation.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server’s heap Measured in Kilobytes (KB) • Response Time The utilization of the server’s allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server’s thread capacity Measured in Threads

	• DBMem The utilization of the database’s memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server’s capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Database Server Targets

This version of Turbonomic supports the following database servers as targets:

- Oracle 11g R2 and 12c
- Microsoft SQL Server 2008 R2, 2012, and 2014
- MySQL 5.5.26 and higher, and all 5.6 releases

You can add an individual database server as a target, or you can add all matching targets within a given group or cluster.

If you set target scope, Turbonomic scans each VM within that group or cluster and tries to connect to a database server over the specified port. Turbonomic adds any database servers it finds as targets.

Supply Chain

Database targets add Application entities to the supply chain. The inventory groups applications by database type — You can navigate to each individual database to see specific details.

Actions

Turbonomic recommends actions for the database supply chain as follows.

NOTE: This is a general list of actions for entities discovered for databases. You can see how actions differ per technology in each section that describes adding a specific type of database target.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
Database Applications	• Resize Database Memory Recommendation only. • Resize Transactions Recommendation only. • Resize Connections Recommendation only.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the database supply chain:

NOTE: This is a general list of commodities for entities discovered for databases. You can see how actions differ per technology in each section that describes adding a specific type of database target.

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Database Applications	• DBMem The memory utilized by the database, as a of the memory capacity that is allocated to the database. Note that this resource is more accurate than the VMem resource on the hosting VM. With this resource, Turbonomic can drive resize and move actions based on the memory consumed by the database, not the memory consumed by the VM

	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server's heap Measured in Kilobytes (KB) • Response Time The utilization of the server's allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server's thread capacity Measured in Threads • DBMem The utilization of the database's memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server's capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS

- Latency
  The utilization of latency allocated for the VStorage on the VM
  Measured in milliseconds (ms)

MySQL

To manage a MySQL databases, Turbonomic can connect:

- To an individual database server
- To all the database servers within a group or cluster

This version of Turbonomic supports MySQL 5.5.26 and higher, and all 5.6 releases.

Prerequisites

- Target is properly configured as Domain Controller or standalone, depending on the operations mode
- User Permissions are enabled on the MySQL Server. See Enabling User Permissions on MySQL Server on page 138

Adding a MySQL Database Target

You can add an individual database server as a target, or you can add all matching targets within a given scope.

To add a database server as a target, you specify:

- **Address:** The name or IP address of the VM hosting the database server
- **Scope (optional):** A cluster or group of VMs that host MySQL servers
  If you set target scope, Turbonomic scans each VM within that group or cluster and tries to connect to a MySQL database server over the specified port. Turbonomic adds any database servers it finds as targets.
- **Port Number:** The port that connects to the MySQL browser
  You must open the firewall on the MySQL server to allow access through this port.
- **Username/Password:** Valid client credentials for the database service
  For Turbonomic to execute actions, the account must have administrator privileges. Also, you must have enabled user permissions to this user account, including remote access from the Turbonomic server.

Actions

Turbonomic recommends actions for the application supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
Database Applications	• Resize Database Memory Recommendation only. • Resize Transactions Recommendation only. • Resize Connections Recommendation only.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Database Applications	• DBMem The memory utilized by the database, as a of the memory capacity that is allocated to the database. Note that this resource is more accurate than the VMem resource on the hosting VM. With this resource, Turbonomic can drive resize and move actions based on the memory consumed by the database, not the memory consumed by the VM • Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB)

	• Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server’s heap Measured in Kilobytes (KB) • Response Time The utilization of the server’s allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server’s thread capacity Measured in Threads • DBMem The utilization of the database’s memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server’s capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Enabling User Permissions on MySQL Server

Follow the following steps in order to enable appropriate user permissions on a MySQL Server.

1. **Edit the MySQL server's configuration file.**

   You must edit the `.conf` file on the MySQL server to grant user permissions. Open a secure shell session on the server and edit the file. Depending on the platform your MySQL is running on, you'll fin the file at different locations:

   - Debian Linux:

     `/etc/mysql/my.cnf`
   - Red Hat Linux (Fedora or Centos):

     `/etc/my.cnf`
   - FreeBSD Linux:

     You must create the file at `/var/db/mysql/my.cnf`

   Open the file in an editor and find the section, `[mysqld]`. Then make the following changes:

   - Comment out the line:

     `skip-networking`

     Commenting out this line enables remote connections over TCP/Is.
   - Add the line

     `bind-address=<MySQL_IP_Address>`

   For example, if your MySQL server has the address, `123.45.66.77`, then the section of the `.conf` file should appear as follows:

   ```
   [mysqld]
   user            = mysql
   pid-file        = /var/run/mysqld/mysqld.pid
   socket          = /var/run/mysqld/mysqld.sock
   port            = 3306
   basedir         = /usr
   datadir         = /var/lib/mysql tmpdir          = /tmp
   language        = /usr/share/mysql/English
   bind-address    = 123.45.66.77 # skip-networking
   ....
   ```

   When you are done, save the `.conf` file.

2. **Give your Turbonomic server remote access to the database.**

   Execute the following command to log into to the MySQL server:

   ```
   $mysql -u root -p mysql
   ```

   Then execute the following command:

   Assume a user named USER_NAME with a password `PWD_STRING`. Then assume that your Turbonomic has an IP address of `10.10.123.45`. The following command grants privileges to that Turbonomic, if it connects with the specified user account:

   ```
               GRANT SELECT ON performance_schema.* TO 'USER_NAME'@'10.10.123.45'
   IDENTIFIED BY 'PWD_STRING';
               GRANT PROCESS ON *.* TO 'USER_NAME'@'10.10.123.45' IDENTIFIED BY
   'PWD_STRING';
               FLUSH PRIVILEGES;
   ```

   When you're finished, log out of MySQL.

Oracle

To connect to an Oracle database, you will:

- Add a Dynamic Performance view to the Oracle database
- Configure a service account on the database that Turbonomic can use to log on
- Find the Service Name (SID) and port for the database

This version of Turbonomic supports Oracle versions 11g R2 and 12c

Prerequisites

- User permissions that grant access to Turbonomic through a specific user account. See Creating a Service User Account in Oracle on page 142.
- Dynamic Performance View `(V$)` must be enabled. See "Adding a Dynamic Performance View", below.
- Access through the firewall to the MySQL database port that you specify for the Turbonomic target connection

Adding a Dynamic Performance View

In order to collect data from the Oracle database, Turbonomic uses the Dynamic Performance View (referred to as V$). V$ is not enabled by default, and users must run a script to build the tables and views that are necessary to enable V$. In some environments only the DBA has privileges to run this script.

To enable V$:

- Open a secure shell session (ssh) on the database host as a system user or a user with the `sysdba` role
- In the shell session enter the following commands:

```
sqlplus /nolog
connect /as sysdba
@?/rdbms/admin/catalog.sql
```

Adding an Oracle Database to Turbonomic

You can add an individual database server as a target, or you can add all matching targets within a given scope.

To add a database server as a target, you specify:

- **Address:** The name or IP address of the VM hosting the database server
- **Scope (optional):** A cluster or group of VMs that host Oracle database servers
  If you set target scope, Turbonomic scans each VM within that group or cluster and tries to connect to an Oracle database server over the specified port. Turbonomic adds any database servers it finds as targets.
- **Port Number:** The port that connects to database SID
  You must open the firewall on the database server to allow access through this port. For further information, see "Finding the Service Name (SID) and Port", below.
- **Service:** The SID for the database that you are connecting to

- **Username/Password:** Valid client credentials for the database server
  For Turbonomic to execute actions, the account must have administrator privileges. Also, you must have enabled user permissions to this user account, including remote access from the Turbonomic server.

Finding the Service Name (SID) and Port

To specify a target, you must provide the SID and port that you want to connect to. To find the SID for your database:

- Open a secure shell session (ssh) on the database host as a system user or a user with the `sysdba` role
- In the shell session, enter the command, `lsnrctl status`
  Find the line that has the string `PROTOCOL=tcp` and note the port number.
- In the shell session enter the following commands:

```
sqlplus /nolog
connect /as sysdba
SELECT SYS_CONTEXT('userenv', 'db_name') FROM dual;
```

  Note the SID that displays as a result of these commands.

Actions

Turbonomic recommends actions for the application supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
Database Applications	• Resize Database Memory Recommendation only. • Resize Transactions Recommendation only. • Resize Connections Recommendation only.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Database Applications	• DBMem The memory utilized by the database, as a of the memory capacity that is allocated to the database. Note that this resource is more accurate than the VMem resource on the hosting VM. With this resource, Turbonomic can drive resize and move actions based on the memory consumed by the database, not the memory consumed by the VM • Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server’s heap Measured in Kilobytes (KB) • Response Time The utilization of the server’s allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server’s thread capacity Measured in Threads • DBMem The utilization of the database’s memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server’s capacity for storage devoted to transaction logs Measured in Kilobytes (KB)

	• Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Creating a Service User Account in Oracle

To collect data from the Oracle database, Turbonomic requires a service account that has privileges to access the V$ Dynamic Performance view. To create this account:

- Open a secure shell session (ssh) on the database host as a system user or a user with the `sysdba` role
- In the shell session enter the following commands:

  ```
  sqlplus /nolog
  connect /as sysdba
  CREATE USER My_Username IDENTIFIED BY My_Password container=all;
  GRANT CONNECT TO My_Username container=all;
  GRANT sysdba TO My_Username container=all;
  ```

  This creates a user account named My_Username with full privileges to access the V$ Dynamic Performance view.

NOTE: The above example uses a fictitious username. To comply with Oracle 12C norms, the username should include a prefix of `C##`.

Some enterprises don’t allow accounts with sysdba access. Turbonomic recommends using sysdba, according to the Oracle documentation. However, you can work with your Oracle DBA staff to provide read access to the following views, which are the ones that Turbonomic needs:

- V$INSTANCE
- V$LOG
- V$LOGFILE
- V$PARAMETER
- V$PGASTAT
- V$RESOURCE_LIMIT
- V$SGASTAT

- V$SYS_TIME_MODEL
- V$SYSMETRIC
- V$SYSSTAT

SQL Server

When connecting to a SQL Server database, you provide a “service” name. This is the name of the actual database, so the target is the database itself, and not a service on the database.

This version of Turbonomic supports Microsoft SQL Server 2008 R2, 2012, and 2014.

Prerequisites

- A user account with SQL permissions on the database
- The following services must be running, and set to enabled:
    - Net.Tcp Listener Adapter
    - Net.Tcp Port Sharing Service
- TC/IP is enabled on the port that you set for the Turbonomic specification

Creating a Service User Account

The user account that Turbonomic uses for its service login must include the following:

- The account must exist in the Security folder within the SQL Server Object Explorer, with the following properties:
    - Enable **SQL Server Authentication**
    - Disable **Enforce password policy**
- The account's security properties must include:
    - Permission to connect to the database through SQL
    - Permission to view the server state

Adding a SQL Server Database to Turbonomic

You can add an individual database server as a target, or you can add all matching targets within a given scope.

To add a database server as a target, you specify:

- Target ID
  The display name that will be used to identify the target in the Target List. This is for display in the UI only; it does not need to match any internal name.
- Username
  The username Turbonomic will use to connect to the Virtual Machine hosting the database
- Password
  The password Turbonomic will use to connect to the Virtual Machine hosting the database

- Secure Connection
  When selected, Turbonomic will connect using SSL
- Domain Name:
  The Active Directory domain used by Turbonomic in conjuction with the Username for authentication. Leave blank for local accounts.
- Scope:
  Turbonomic will search for SQL Servers on the VMs found within the set scope. This scope can be set to the entire environment, single or multiple clusters, or particular physical machines. Select this option and choose the scope for application discovery.

For additional information about adding SQL Server targets, see the Green Circle article, "How to Target SQL Servers in VMTurbo (5.4 and later)"

Actions

Turbonomic recommends actions for the application supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
Applications	Without discovered Guest OS Processes, Turbonomic doesn’t perform actions on applications. Instead, it performs resize actions on the host VMs. If host utilization is high enough on the physical machine running the application VM, Turbonomic may also recommend provisioning a new host. For specific applications, see the individual application entry.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Database Applications	• DBMem The memory utilized by the database, as a of the memory capacity that is allocated to the database. Note that this resource is more accurate than the VMem resource on the hosting VM. With this resource, Turbonomic can drive resize and move actions based on the memory consumed by the database, not the memory consumed by the VM • Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server’s heap Measured in Kilobytes (KB) • Response Time The utilization of the server’s allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server’s thread capacity Measured in Threads • DBMem The utilization of the database’s memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server’s capacity for storage devoted to transaction logs Measured in Kilobytes (KB)

	• Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Windows® Application Targets

This version of Turbonomic supports adding Microsoft® Exchange® servers as targets. Turbonomic can monitor the application server resources and recommend actions to scale server capacity horizontally (provision new servers) or vertically (resize existing servers).

You can add an individual Exchange server as a target, or you can add all matching targets within a given scope.

Prerequisites

- Configure remote management on the Exchange server. Refer to the relevant section below:
  - **Turbonomic** 5.4 or higher: See Creating A Service User Account on page 17
  - **Turbonomic** 5.3 or below: See Enabling Management via DCOM on page 189
- Port access

  WinRM uses ports 5985 and 5986 for standard and secure communications, respectively. The firewall on the your VMM server must open these ports.

Supply Chain

Exchange Server targets add Application entities to the supply chain. The inventory groups applications by server type — You can navigate to each individual application server to see specific details.

Application Server Scaling Policy

After you specify an application server target, Turbonomic discovers the resource utilization for that server, and recommends appropriate actions. These actions will follow one of the two following scaling policies:

- Provision
  Scaling by Provision enables horizontal scaling, where the environment adjusts to increased demand by provisioning new application servers.
- Resize
  Scaling by Resize enables vertical scaling, where the environment adjusts to increased demand by resizing the application server, or the VM that hosts the application server.

These scaling policies are mutually exclusive — the affected application servers can scale by Provisioning or by Resizing, but not both. For example, Turbonomic can recommend resize actions, but if you have set the Scaling Policy to Provision, Turbonomic will only recommend Provision and Suspend actions.

You set the Scaling Policy in the Policy view — Turbonomic policies support scope, so you can select specific groups of application servers to set their Scaling Policies as you like. For more information, see "Application Server Actions" in the Turbonomic User Guide.

Adding Microsoft Exchange Server Targets

Once you’ve enabled remote management, you can add your Exchange servers as targets. To add an Exchange Server target, select the **Windows Application > MS Exchange** option on the Target Configuration page and provide the following information:

- Host
  The host name of the Exchange server.
  Note that you can enter an IP address for the host, but you must first configure an SPN on the host. Turbonomic recommends that you use the host name in this field.
- Scope (optional)
  A cluster or group of VMs that host Exchange servers
  If you set target scope, Turbonomic scans each VM within that group or cluster and tries to connect to an Exchange server over the specified port. Turbonomic adds any servers it finds as targets.
- Port number
  For Turbonomic 5.4 or higher: The port number for the remote management connection. The default HTTP port is 5985; the default HTTPS port is 5986.
- Username
  The domain\username of an Active Directory account that Turbonomic can use to connect to the Exchange server.
  For Turbonomic 5.3 or below: The account must also be one of the server’s WBEM Scripting Locator owners (see Enabling Management via DCOM on page 189).
- Password
  Password for account used.

Actions

Turbonomic recommends actions for the application supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
Applications	Without discovered Guest OS Processes, Turbonomic doesn’t perform actions on applications. Instead, it performs resize actions on the host VMs. If host utilization is high enough on the physical machine running the application VM, Turbonomic may also recommend provisioning a new host. For specific applications, see the individual application entry.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the application server supply chain:

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz)

	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server’s heap Measured in Kilobytes (KB) • Response Time The utilization of the server’s allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server’s thread capacity Measured in Threads • DBMem The utilization of the database’s memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server’s capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Network Flow Targets

With Network Targets, Turbonomic can calculate costs associated with network proximity when managing workload placement. For example, two VMs that show a lot of network traffic between each other should be placed close together. They would see the best network performance if they were placed on the same physical host, and latency would increase as traffic has to hop more switches to pass data between them. With a Network Target added, Turbonomic can include the benefits of localizing network traffic in its analysis.

For network analysis, Turbonomic groups entities into two types of group:

- VPods
  Sets of consumers that communicate frequently with each other over the network — for example, VMs that run processes for the same distributed application.
- DPods
  Sets of closely connected providers (providers underneath the same switch) — For example a storage controller, its datastores, and the hosts that consume the storage resources. A unified fabric chassis can also make up a DPod, as can the providers under a switch in an Arista network.

In this scheme, DPods provide network flow to VPods, and VPods provide this flow to their constituent consumers. For example, as Turbonomic calculates the placement of a VM on a host or datastore, it considers the cost of the network flow so that VMs can reside closer together if that will lower the overall cost of their placement.

There are four levels of cost for network flow:

- Zero
  The consumers use the same provider — For example VMs that reside on the same host. Such consumers have infinite net throughput capacity.
- Low
  The consumers reside within the same DPod. Net throughput capacity for these consumers is determined by the capacity of the providers on the DPod.
- Medium
  The consumers communicate across DPods (communicate through multiple switches). Throughput capacity is the capacity of the uplink, divided by the number of providers sharing it.
- High
  The consumers communicate across the cloud.

Turbonomic takes these costs into account when making workload placement decisions.

License Requirements

- Network targets are available in the Premier Edition of Turbonomic
- To discover DPods via storage controllers, a storage target
- To discover DPods via UCS fabrics, a fabric target

Supply Chain

Network targets add VPod and DPod entities to the Supply Chain.

Actions

Turbonomic recommends actions for the network target supply chain as follows.

NOTE: This is a general list of actions for entities discovered for network targets. You can see how actions differ per technology in each section that describes adding a specific type of network target.

Entity Type	Action
VPod	• Move VPod across DPod • Move VM across VPods
DPod	• Provision new DPod

Monitored Resources

Turbonomic monitors the following resources for the network supply chain:

NOTE: This is a general list of commodities for entities discovered for network targets. You can see how actions differ per technology in each section that describes adding a specific type of network target.

Entity Type	Commodity
VPod	• Memory (Mem) The utilization of the VPod's memory reserved or in use Measured in Kilobytes (KB)

	• CPU The utilization of the VPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the VPod Measured in Kilobytes (KB) • Flow The utilization of the network flow capacity utilized by the VPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization Measured in Kilobytes per second (KB/s)
DPod	• Memory (Mem) The utilization of the DPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the DPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the DPod Measured in Kilobytes (KB) • Flow The utilization of the network flow capacity utilized by the DPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization Measured in Kilobytes per second (KB/s)

Arista

To monitor an Arista network, choose a single Top-Of-Rack switch from the network you want to monitor and add it as a Turbonomic target. Turbonomic can discover the rest of the network through that single switch. As a result of discovery, Turbonomic builds one DPod per switch.

note: An Arista target only discovers DPods from the Arista switches. To include VPods in your network analysis, you must also add flow connectors. For more information, see NetFlow on page 154 or sFlow on page 156.

Prerequisites

- The Arista Command eAPI must be enabled for HTTPS communication (see the Arista documentation for details)
- A service account Turbonomic can use to connect to Arista. This account must be able to connect to the Arista eAPI.

Adding Arista Targets to Turbonomic

To add an Arista target, select the **Network > Arista** option on the Target Configuration page and provide the following information:

- **Address:** The IP address of the top-of-rack switch
- **Username/Password:** The credentials of an account Turbonomic can use to connect to the Arista switch.

Actions

Turbonomic recommends actions for the network target supply chain as follows.

Entity Type	Action
DPod	• Provision new DPod

Monitored Resources

Turbonomic monitors the following resources for the network supply chain:

Entity Type	Commodity
DPod	• Memory (Mem) The utilization of the DPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the DPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the DPod Measured in Kilobytes (KB) • Flow The utilization of the network flow capacity utilized by the DPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization Measured in Kilobytes per second (KB/s)

NetFlow

Turbonomic can connect to NetFlow data collectors to gather information about traffic between VMs, hosts, and storage. Using this information, Turbonomic can build VPods and make recommendations that reduce network latency.

NOTE: The NetFlow target builds VPods and includes them in Turbonomic analysis. To get the most benefit from network control, you should also build DPods. To do this, you must add targets for Arista networks, storage controllers, or fabrics.

For information about adding these targets, see:

- Arista on page 153
- Storage Targets on page 66
- "Adding Fabric Targets" in the Turbonomic User Guide

Prerequisites

- A NetFlow data collector
- A service account Turbonomic can use to connect to the NetFlow data collector
  The service account Turbonomic uses must have permission to remotely SSH into the data collector and run NFDUMP.

NetFlow Data Collector Requirements

The NetFlow data collector that you use must include NFDUMP tools, and the `nfcapd` service (NetFlow capture daemon) must be running on the NetFlow system.

Turbonomic uses `nfdump` to import the NetFlow data from the collector.

Turbonomic provides a NetFlow data collector OVA with NFDUMP preconfigured.

For more information and to download this OVA, see the Green Circle article: NetFlow and sFlow Instrumentation and Monitoring.

To verify that the data collector is capturing data, open a secure shell on the collector’s host, log in using the service account that you specify for the Turbonomic target, and execute the following command:

```
nfdump -R /var/netflow
```

vSphere Distributed Switch Requirements

If you are collecting NetFlow data from a vSphere Distributed Switch (vDS):

- The vDS must be configured to point to the NetFlow data collector
- NetFlow Monitoring must be enabled on the distributed port group associated with each vDS

For information about these configuration steps, refer to your vSphere documentation.

Adding NetFlow Targets to Operations Manager

To add a NetFlow target, select the **Network > NetFlow** option on the Target Configuration page and provide the following information:

- **Address:** The IP address of the Netflow data collector
- **Username/Password:** The credentials of an account Turbonomic can use to connect to the data collector.

Actions

Turbonomic recommends actions for the network target supply chain as follows.

Entity Type	Action
VPod	• Move VPod across DPod • Move VM across VPods

Monitored Resources

Turbonomic monitors the following resources for the network supply chain:

Entity Type	Commodity
VPod	• Memory (Mem) The utilization of the VPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the VPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the VPod Measured in Kilobytes (KB) • Flow The utilization of the network flow capacity utilized by the VPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization Measured in Kilobytes per second (KB/s)

sFlow

Turbonomic can connect to SFlow data collectors to gather information about traffic between VMs, hosts, and storage. Using this information, Turbonomic can build VPods and make recommendations that reduce network latency.

NOTE: The sFlow target builds VPods and includes them in Turbonomic analysis. To get the most benefit from network control, you should also build DPods. To do this, you must add targets for Arista networks, storage controllers, or fabrics.

For information about adding these targets, see:

- Arista on page 153

- Storage Targets on page 66
- Fabric Targets on page 166

Prerequisites

- An sFlow data collector

  You must have an sFlow data collector running on your network. Make sure the `sflow-rt` service is running. Turbonomic connects to the data collector through its REST API on port 8008. Credentials are not typically required to access the SFlow API. If you require a secure connection, contact Turbonomic Technical Support for assistance.

  Turbonomic provides an OVA with a preconfigured sFlow-RT data collector.

  For more information and to download this OVA, see the Green Circle article: NetFlow and sFlow Instrumentation and Monitoring.

Adding sFlow Targets to Operations Manager

To add a NetFlow target, select the **Network > sFlow** option on the Target Configuration page and provide the following information:

- **Address:** The IP address of the sFlow data collector
- **Username/Password:** The credentials of an account Turbonomic can use to connect to the data collector.

Actions

Turbonomic recommends actions for the network target supply chain as follows.

Entity Type	Action
VPod	• Move VPod across DPod • Move VM across VPods

Monitored Resources

Turbonomic monitors the following resources for the network supply chain:

Entity Type	Commodity
VPod	• Memory (Mem) The utilization of the VPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the VPod's CPU reserved or in use Measured in Megahertz (MHz)

- Storage
  The utilization of the storage attached to the VPod
  Measured in Kilobytes (KB)
- Flow
  The utilization of the network flow capacity utilized by the VPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization
  Measured in Kilobytes per second (KB/s)

Cisco APIC

Turbonomic uses TCAM (Ternary Content-Addressable Memory) table utilization to make network-aware VM placement decisions under top-of-rack APIC (Application Policy Infrastructure Controller) switches in such a way that the tables remain balanced. To monitor a network using Cisco APIC, add it as a Turbonomic target. Turbonomic will discover the TCAM utilization of the underlying infrastructure.

NOTE: Adding a Cisco APIC target will not cause any service entities to appear in the market. Instead, Turbonomic represents TCAM utilization using a commodity that Physical Machines under the APIC sell to the associated VMs.

Prerequisites

- A service account Turbonomic can use to connect to Cisco APIC. Unlike most other targets, this account can be read-only as Turbonomic does not make action recommendations directly through the Cisco APIC.

Adding Cisco APIC Targets to Turbonomic

To add a Cisco APIC target, select the **Network > Cisco APIC** option on the Target Configuration page and provide the following information:

- **Address:** The IP address of the Cisco APIC.
- **Username/Password:** The credentials of an account Turbonomic can use to connect to the Cisco APIC.

Actions

The commodity that represents TCAM utilization assists in driving actions in the market, in conjunction with other factors such as resource usage, etc.

Application Container Targets

If you are running the Cloud Native Edition of Turbonomic, it will discover application containers in your environment, recommend resize actions, and display discovered containers in the Inventory view. This release of Turbonomic supports Docker containers.

To use this functionality, you must specify a container target. These targets are found under the "Application Server" target type in the Target Configuration section of the Admin tab.

Supply Chain

Turbonomic adds Container objects to the inventory. A container object hosts an application.

Actions

Turbonomic recommends actions for the container supply chain as follows.

Entity Type	Action
Docker Container	Resize Container Recommend Only
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the container supply chain:

Entity Type	**Commodity**
ContainerPod	• VMem The of memory capacity the container utilizes Measured in Kilobytes (KB) • VCPU The of CPU capacity the container utilizes Measured in Megahertz (Mhz)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Setting Up Application Containers Managed By Kubernetes

Turbonomic will discover application containers running across all pods in a Kubernetes cluster.

Prerequisites

- Turbonomic 5.9+ installation
- Running Kubernetes 1.4+ cluster

Adding Kubernetes Targets

In order to add this target, you must create a Kubeturbo configuration file with your Turbonomic credentials and a custom pod definition that will be used by Kubelet to create a mirror pod running the Kubeturbo service.

For more information, please see the Kubeturbo github repository, located at https://github.com/turbonomic/kubeturbo and specifically, the readme located at https://github.com/turbonomic/kubeturbo/blob/master/deploy/general-deploy/README.md

NOTE: You must install Kubeturbo on each cluster you want Turbonomic to discover. In order to execute actions, the default scheduler must be replaced. This process is outlined in the repository readme.

Supported Actions

For each discovered entity within the container supply chain, Turbonomic can execute or recommend certain actions, as outlined below.

For actions to be executed, the default scheduler must be replaced.

Entity Type	Action
Kubernetes ContainerPod	• Horizontal Scale Up • Horizontal Scale Down • Provision Additional Resources (VMem, VCPU) Recommend Only • Move Pod across VM

Monitored Resources

Turbonomic monitors the following resources for the container supply chain:

Entity Type	Commodity
ContainerPod	• VMem The of memory capacity the container utilizes Measured in Kilobytes (KB) • VCPU The of CPU capacity the container utilizes Measured in Megahertz (Mhz)

PaaS Manager Targets

Platform as a Service (PaaS) is a management model to deliver applications over the internet. A PaaS provider delivers hardware and software tools as a service, and customers use the service to build and deploy applications.

Turbonomic supports the CloudFoundry platform. For targets, you can specify PaaS implementations that use the CloudFoundry SDK v.2. Turbonomic has tested implementations of HP Stackato, but Turbonomic should support other implementations on v.2, such as Pivotal.

Identifying the CloudFoundry API Endpoint

To specify a target, you attach to a CloudFoundry API endpoint. The following steps describe how to identify the endpoint in HP Stackato:

- Open the Management Console
- Navigate to **Admin > Cluster Management**
- Find the entry for **API Endpoint** — That shows the IP address you need for the Turbonomic target

Adding CloudFoundry Targets

To add CloudFoundry targets, select the **PaaS > CloudFoundry** option on the Target Configuration page and provide the following information:

- Address
  The address of the API endpoint.
- Username/Password
  Credentials for an account on the endpoint with administrator privileges.

Actions

Turbonomic recommends actions for the PaaS supply chain as follows.

Entity Type	Action
Applications	Without discovered Guest OS Processes, Turbonomic doesn’t perform actions on applications. Instead, it performs resize actions on the host VMs. If host utilization is high enough on the physical machine running the application VM, Turbonomic may also recommend provisioning a new host. For specific applications, see the individual application entry.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Physical Machines	• Start Physical Machine • Provision Physical Machine • Suspend Physical Machine
Storage	• Start Storage • Provision Storage • Suspend Storage • Move (only with Storage Targets configured) • Resize (only with Storage Targets configured)

Monitored Resources

Turbonomic monitors the following resources for the PaaS supply chain:

Entity Type	Commodity
Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second

	• Heap The utilization of the application server’s heap Measured in Kilobytes (KB) • Response Time The utilization of the server’s allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server’s thread capacity Measured in Threads • DBMem The utilization of the database’s memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server’s capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s)

	• Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)

Fabric Targets

A fabric target is a service that unites compute, network and storage access into a cohesive system. When you connect Turbonomic to fabric targets, it monitors the performance and resource consumption of your fabric interconnects, IO modules, chassis, and Blade Servers to assure application performance while also utilizing resources as efficiently as possible.

Once connected, Turbonomic discovers the blades that host the VMs, the chassis and datastores that provide resources to the blades, the IO modules and fabric interconnects that provide network resources, and the virtual datastores that provide storage resources to the VMs.

As part of this process, Turbonomic will stitch information from the fabric target and the associated hypervisor targets together, to provide deeper insight into the state of the hardware, and information related to the Applications and VM's running on the hypervisior-stitched blades.

License Requirements

Fabric targets are supported by all Editions of Turbonomic.

Supply Chain

Fabric targets add IO Module, Fabric Interconnect, Domain, and Chassis entities to the supply chain. The Chassis entities host physical machines (blade servers) — The physical machines also consume network connection commodities from IO Modules. The Fabric Interconnect supplies connectivity to the overall network, and also hosts the UCS Manager. The Domain serves as the bottom-level pool of network resource, supplying the Fabric Interconnect.

Supported Fabric Targets

Turbonomic supports the following fabric targets:

- Cisco UCS
- Cisco UCS Central
- HP Oneview

Monitored Resources

Turbonomic monitors the following resources for the fabric supply chain:

Entity Type	**Commodity**
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Blade	• Net The utilization of data through the Blade's network adapters Measured in Kilobytes per second (KB/s) • Treated as a Physical Machine of the underlying Hypervisor (see below) CPU, Mem, etc.
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s)

	• Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
I/O Module	• NetThroughput Rate of message delivery over a port Measured in Megabits per second (Mb/s)
Fabric Interconnect	• NetThroughput Rate of message delivery over a port Measured in Mb/s • PortChannel Amalgamation of ports with a shared net throughput and utilization Measured in Mb/s

Cisco UCS Manager

The Cisco Unified Computing System (UCS) Manager is a management solution that participates in server, fabric, and storage provisioning, device discovery, inventory, configuration, diagnostics, monitoring, fault detection, auditing, and statistics collection.

UCS integrates all of these resources in a scalable multi-chassis platform to converge administration onto a single point. Managing these various entities on a network fabric with Turbonomic enables automation at the hardware level, including automated provisioning of hosts.

Prerequisites

- A service account Turbonomic can use to connect to UCS Manager

Adding UCS Targets

To add a UCS target, select the **Fabric** category and choose one of the UCS Fabric options to match the version of UCS you want to manage. Then provide the following information:

- **Address:** The IP address of the UCS Manager

  This gives access to the Fabric Manager that resides on the interconnect.

  Turbonomic connects to the UCS Manager via the HTTPS protocol by default. In order to force the HTTP protocol, the Address must be entered in one of two ways. For example, an IP of 8.8.8.8 must be entered as http://8.8.8.8 or by using a specific HTTP port, such as 8.8.8.8:80.
- **Username/Password:** The credentials of an account Operations Manager can use to connect to UCS Manager.

  specify the IP address and credentials for UCS Manager. Turbonomic discovers the fabric interfaces associated with that manager.

NOTE: When providing a username, if the account is managed in Active Directory you must include the domain in case-sensitive spelling. For example,`MyDomain\john` is not the same as `mydomain\john`. For local user accounts, just provide the username.

Supply Chain

Fabric targets add IO Module, Fabric Interconnect (Switch), Domain, and Chassis entities to the supply chain. The Chassis entities host physical machines — The physical machines also consume network connection commodities from IO Modules. The Fabric Interconnect supplies connectivity to the overall network, and also hosts the HPE OneView. The Domain serves as the bottom-level pool of network resource, supplying the Fabric Interconnect.

Actions

Turbonomic recommends actions for the various entities of the UCS Fabric Network as follows:

Entity Type	Action
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Physical Machines	• Start Physical Machine • Provision Physical Machine • Suspend Physical Machine
Chassis	• Provision New Chassis
Fabric Interconnect	• Add Port to Port Channel • Remove Port from Port Channel • Add Port
DPod	• Provision new DPod

Monitored Resources

Turbonomic monitors the following commodities of the UCS target:

Entity Type	**Commodity**
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)

Chassis	• Space Representation of the physical number of Rack-Mounts available Measured in Rack-slots • Power Electricity being consumed by the Chassis Measured in Watts (W) • Temperature Temperature of the internals of the Chassis Measured in degrees Celsius (C)
I/O Module	• NetThroughput Rate of message delivery over a port Measured in Megabits per second (Mb/s)
Fabric Interconnect	• NetThroughput Rate of message delivery over a port Measured in Mb/s • PortChannel Amalgamation of ports with a shared net throughput and utilization Measured in Mb/s
DPod	• Memory (Mem) The utilization of the DPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the DPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the DPod Measured in Kilobytes (KB) • Flow The utilization of the network flow capacity utilized by the DPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization Measured in Kilobytes per second (KB/s)

Cisco UCS Central

Cisco UCS Central aggregates multiple Cisco Unified Computing System targets onto a single point of management. Managing compute resources on a network fabric of this magnitude enables automation at the hardware level, including automated provisioning of hosts on any of the underlying UCS Manager instances.

Turbonomic does not discover entities through the UCS Central target. Instead, all entities are identified from the UCS Manager instances associated to the UCS Central Software, and discovered through the UCS Managers.

Prerequisites

- A service account Turbonomic can use to connect to UCS Central Software

Adding UCS Central Targets

To add a UCS Central target, select the **Fabric** category and choose UCS Central. Then provide the following information:

- **Address:** The IP address of the Virtual Machine running the UCS Central OVA.

  This gives access to the UCS Central Software that resides on the Virtual Machine.

  Turbonomic uses the HTTPS protocol by default. In order to force the HTTP protocol, the address must be entered in one of two ways. For example, an IP of 8.8.8.8 must be entered as http://8.8.8.8 or by using a specific HTTP port, such as 8.8.8.8:80.
- **Username/Password:** The credentials of an account Turbonomic can use to connect to UCS Central Software.

  Specify the IP address and credentials for UCS Central Software. Turbonomic discovers the UCS Managers associated with the instance of UCS Central.

NOTE: When providing a username, if the account is managed in Active Directory you must include the domain in case-sensitive spelling. For example, `MyDomain\john` is not the same as `mydomain\john`. For local user accounts, just provide the username.

After you add a UCS Central target, Turbonomic will discover all the associated UCS Manager targets that are part of the UCS Central instance.

Supply Chain

Fabric targets add IO Module, Fabric Interconnect (Switch), Domain, and Chassis entities to the supply chain. The Chassis entities host physical machines — The physical machines also consume network connection commodities from IO Modules. The Fabric Interconnect supplies connectivity to the overall network, and also hosts the HPE OneView. The Domain serves as the bottom-level pool of network resource, supplying the Fabric Interconnect.

Actions

Turbonomic recommends actions for the various UCS Manager instances of the UCS Central Fabric Network as follows:

Entity Type	Action
Fabric Interconnect	• Add Port to Port Channel Overutilization of Ports in Port Channel • Remove Port from Port Channel Underutilization of Ports in Port Channel • Add Port Overutilization of Ports not in Port Channels

Entity Type	Action
Blade	• Provision Blade Overutilization of currently available Blades Available powered-down Blade without associated host profile • Start Blade Overutilization of currently available blades Available powered-down Blade with associated host profile • Suspend Blade Underutilization of currently available Blades

NOTE: Turbonomic will also recommend adding a port to the I/O Module via an "Add Port to Port Channel" recommendation from the Fabric Interconnect, with the Port Channel name of "fex" (Fabric EXtender). Turbonomic will never resize down the I/O module Port Channel.

Monitored Resources

Turbonomic monitors the following commodities of each underlying UCS Manager instance:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)
Blade	• Net The utilization of data through the Blade's network adapters Measured in Kilobytes per second (KB/s) • Treated as a Physical Machine of the underlying Hypervisor (see below) CPU, Mem, etc.

Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Chassis	• Space Representation of the physical number of Rack-Mounts available Measured in Rack-slots • Power Electricity being consumed by the Chassis Measured in Watts (W) • Temperature Temperature of the internals of the Chassis Measured in degrees Celsius (C)
I/O Module	• NetThroughput Rate of message delivery over a port Measured in Megabits per second (Mb/s)
Fabric Interconnect	• NetThroughput Rate of message delivery over a port Measured in Mb/s • PortChannel Amalgamation of ports with a shared net throughput and utilization Measured in Mb/s

HPE OneView

HPE OneView is a management solution that streamlines provisioning and lifecycle management across compute, storage, and fabric. Through a unified API, infrastructure can be configured, monitored, updated, and repurposed.

HP OneView integrates all of these resources in a scalable multi-enclosure platform to converge administration onto a single point. Managing these various entities on a network fabric with Turbonomic enables automation at the hardware level, including automated provisioning of hosts.

HPE OneView is a management solution that streamlines provisioning and lifecycle management across compute, storage, and fabric. Through a unified API, infrastructure can be configured, monitored, updated, and repurposed.

Prerequisites

- A service account Turbonomic can use to connect to HPE OneView.
- HPE OneView 2.0 and compatible hardware.

Adding HPE OneView Targets

To add a HPE OneView as a target, select the **Fabric** category and choose the HPE OneView radio button. Then provide the following information:

- **Address:** The IP address of the HPE OneView target

  This gives access to the Fabric Manager that resides on the VM.

  Turbonomic uses the HTTPS protocol by default. In order to force the HTTP protocol, the Address must be entered in one of two ways. For example, an IP of 8.8.8.8 must be entered as http://8.8.8.8 or by using a specific HTTP port, such as 8.8.8.8:80.
- **Username/Password:** The credentials of an account Operations Manager can use to connect to the HPE OneView target.

  specify the IP address and credentials for HPE OneView. Turbonomic discovers the fabric interfaces associated with that instance.

NOTE: When providing a username, if the account is managed in Active Directory you must include the domain in case-sensitive spelling. For example, `MyDomain\john` is not the same as `mydomain\john`. For local user accounts, just provide the username.

Supply Chain

Fabric targets add IO Module, Fabric Interconnect (Switch), Domain, and Chassis entities to the supply chain. The Chassis entities host physical machines — The physical machines also consume network connection commodities from IO Modules. The Fabric Interconnect supplies connectivity to the overall network, and also hosts the HPE OneView. The Domain serves as the bottom-level pool of network resource, supplying the Fabric Interconnect.

NOTE: For HPE OneView targets, the "Fabric Interconnect" entity exists as a false "Switch", and only as a pass-through for network resources. Unlike other fabric targets, such as UCS, there is no physical hardware that serves this function.

Actions

Turbonomic recommends actions for the various entities of the HPE OneView Fabric Network as follows:

Entity Type	Action
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine
Physical Machines	• Start Physical Machine • Provision Physical Machine • Suspend Physical Machine
Fabric Interconnect	• Add Port to Port Channel • Remove Port from Port Channel • Add Port
DPod	• Provision new DPod

Monitored Resources

Turbonomic monitors the following commodities of the HPE OneView target:

Entity Type	Commodity
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Physical Machine	• Memory (Mem) The utilization of the PM's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the PM's CPU reserved or in use Measured in Megahertz (MHz) • IO The utilization of the PM's IO adapters Measured in Kilobytes per second (KB/s) • Net The utilization of data through the PM's network adapters Measured in Kilobytes per second (KB/s) • Swap The utilization of the PM's swap space Measured in Kilobytes (KB) • Balloon The utilization of shared memory among VMs running on the host. ESX-only Measured in Kilobytes (KB) • CPU Ready The utilization of the PM’s allocated ready queue capacity (measured in Kbytes) that is in use, for 1, 2, and 4 CPU ready queues. ESX-only Measured in Megahertz (MHz)
Storage	• Storage Amount The utilization of the datastore's capacity Measured in Megabytes (MB) • Storage Provisioned The utilization of the datastore's capacity, including overprovisioning. Measured in Megabytes (MB) • Storage Access Operations Per Second (IOPS) The utilization of the storage access operations per second on the datastore Measured in Operations per second • Latency The utilization of latency on the datastore Measured in Milliseconds (ms)
I/O Module	• NetThroughput Rate of message delivery over a port Measured in Megabits per second (Mb/s)
Fabric Interconnect	• NetThroughput Rate of message delivery over a port Measured in Mb/s • PortChannel Amalgamation of ports with a shared net throughput and utilization Measured in Mb/s

DPod	• Memory (Mem) The utilization of the DPod's memory reserved or in use Measured in Kilobytes (KB) • CPU The utilization of the DPod's CPU reserved or in use Measured in Megahertz (MHz) • Storage The utilization of the storage attached to the DPod Measured in Kilobytes (KB) • Flow The utilization of the network flow capacity utilized by the DPod. This is divided into Flow1 (Low Cost) and Flow2 (Medium Cost) utilization Measured in Kilobytes per second (KB/s)

UCS Director

supports dynamic execution of UCS Director workflows, enabling provisioning, resizing and decommissioning of entities in a UCS environment.

Prerequisites

- UCS Director 5.0+
- UCS Director user account with administrator permissions
- UCS Director Workflow Configuration (see below)

NOTE: If UCS Director is managing other services which are also targets (for example, NetApp or vCenter), these targets must be added using the same convention (Hostname or IP) in both UCS Director and .

Adding UCS Director Targets

To add UCS Director targets, select the **Orchestration > UCS Director** option on the Target Configuration page and provide the following information:

- Address
  IP of the UCS Director Instance
- Username
  Username for the UCS Director Instance
- Password
  Password for the UCS Director Instance
- Port Number
  Port Number used to access the UCS Director Instance

Binding A UCS Director Workflow to a Action Execution Policy

NOTE: Orchestrator workflows can only be associated to action execution in the HTML5 UI.

To add a UCS Director workflow, perform the following steps:

1. **Navigate to Settings > Policies**, and select the **Automation Policy button.**
2. **Select the entity that the workflow affects, bringing you to the policy configuration page.**
3. **Under Action Orchestration, select the action that the workflow affects.**
4. **In the Run In dropbox that appears after the action is selected, choose UCS Director.**
5. **Click Link Workflow and select the applicable workflow.**
6. **Click Save and Apply. The workflow is now activated.**

Supported Workflows

To execute each workflow, will provide a set number of pre- defined parameters to Turbonomic as outlined below. Your workflow must be designed to accept the parameters mentioned below for integration to succeed. No parameters may be omitted, or integration will fail.

Workflow discovery will occur once per day. To discover new workflows sooner, you must manually rediscover the UCS Director target.

NOTE: For example workflows that leverage these parameters, please see the following Cisco documentation: https://communities.cisco.com/docs/DOC-75351.

- Host Provision

NOTE: The intent behind this workflow is to provide a user with the information they need to enable a new UCS Server and to configure a hypervisor on that server.

Parameter	Description	Sample Output
POD	The name of the Pod	Default Pod
SERVER_FILER	The UCSM IP/FQDN of the host	ucsmanager.domain.com
SERVER	The stitched IP/FQDN of the host	vcenter1.domain.com
REFERENCE_HOST	The name of the reference host	hypervisor.domain.com
HOST_PROFILE	The name of the host profile	vcenter1@HostProfileName
SERVICE_PROFILE_TEMPLATE_ID	The ID of the service profile template	ucspe;org-root;org-root/ls-sp1_template
SERVICE_PROFILE_ORGANIZATION	The name of the service profile organization	org-root

Parameter	Description	Sample Output
HV_ACCOUNT	The account name of the hypervisor target	vcenter1
REFERENCE_SERVER	The name of the UCS Manager reference server	sys/chassis-1/blade-1
UCSM_ACCOUNT	The name of the UCS Manager account	ucsmanager
DATACENTER	The name of the datacenter for the new host	DataCenter1
CLUSTER	The name of the cluster for the new host	Cluster-1

- Host Suspend

Parameter	Description	Sample Output
POD	The name of the Pod	Default Pod
SERVER_FILER	The UCSM IP/FQDN of the host	ucsmanager.domain.com
SERVER	The stitched IP/FQDN of the host	vcenter1.domain.com
HOST	The host name to suspend	hypervisor.domain.com
HV_ACCOUNT	The account name of the hypervisor target	vcenter1

- Storage Resize

Parameter	Description	Sample Output
POD	The name of the Pod	Default Pod
SERVER_FILER	The UCSM IP/FQDN of the storage	netapp-cluster1.domain.com
SERVER	The stitched IP/FQDN of the hypervisor	vcenter1.domain.com
SC_DATASTORE	The name of the datastore in the Storage Controller	ESXvol1
SC_ACCOUNT	The account name of the storage controller target	netapp-cluster1
HV_DATASTORE	The names of the datastores in the Hypervisor	NETAPP8.3-VOL1
HV_ACCOUNTS	The account names of the Hypervisor targets	vcenter1
SIZE	The target size of the storage in GB	3.75
HOSTS	The hosts that the datastore is exposed to	hypervisor1.domain.com, hypervisor2.domain.com, hypervisor3.domain.com
LUN_NAME	The LUN to resize on the Storage Target	NETAPP8.3-LUN1

- Storage Delete

Parameter	Description	Sample Output
POD	The name of the Pod	Default Pod
SERVER_FILER	The UCSM IP/FQDN of the storage	netapp-cluster1.domain.com
SERVER	The stitched IP/FQDN of the hypervisor	vcenter1.domain.com

Parameter	Description	Sample Output
SC_DATASTORE	The name of the datastore in the Storage Controller	ESXvol1
SC_ACCOUNT	The account name of the storage controller target	netapp-cluster1
HV_DATASTORE	The names of the datastores in the Hypervisor	NETAPP8.3-VOL1
HV_ACCOUNTS	The account names of the Hypervisor targets	vcenter1
HOSTS	The hosts that the datastore is exposed to	hypervisor1.domain.com, hypervisor2.domain.com, hypervisor3.domain.com
LUN_NAME	The LUN to resize on the Storage Target	NETAPP8.3-LUN1

- Storage Provision

Parameter	Description	Sample Output
POD	The name of the Pod	Default Pod
SERVER_FILER	The UCSM IP/FQDN of the storage	netapp-cluster1.domain.com
SERVER	The stitched IP/FQDN of the hypervisor	vcenter1.domain.com
SC_REFERENCE_ DATASTORE	The name of the reference datastore in the Storage Controller	ESXvol1
SC_ACCOUNT	The account name of the storage controller target	netapp-cluster1
DATASTORE_PROVIDER	The provider (DiskArray / LogicalPool) of the new datastore	netapp-cluster1:aggr1
HOSTS	The hosts that the datastore is exposed to	hypervisor1.domain.com, hypervisor2.domain.com, hypervisor3.domain.com
HV_ACCOUNTS	The account names of the Hypervisor targets	vcenter1
SIZE	The target size of the storage in GB	3.75
LUN_NAME	The LUN to resize on the Storage Target	NETAPP8.3-LUN1

Load Balancer Targets

A load balancer is a network appliance that provides Level 4 load balancing of requests to application services. Level 4 load balancing routes traffic according to business rules so that servers with the least load or fastest response times will receive requests from the load balancer.

Turbonomic supports Citrix NetScaler load balancers.

If you have installed the Application Edition of Turbonomic, you can add load balancers as targets. When you add these targets, Turbonomic can discover the associated virtual applications.

Turbonomic recommends actions on the VMs that host the underlying applications. If the VM hosts an application with a signature that is associated with a virtual application, then provisioning the new VM will automatically bind the underlying application to the virtual application. In this way, Turbonomic can control the underlying resources that support the load balancer’s demand.

Supply Chain

A load balancer target adds virtual applications to the supply chain. Turbonomic discovers the virtual applications (sometimes called virtual servers) that are managed by that load balancer. The virtual application is a proxy for multiple instances of actual applications. When a client requests application services, it requests them through the virtual application. The load balancer distributes the requests to balance the workload on the underlying applications that provide transactions to the virtual application.

Configuring Load Balancer Discovery

After you add a load balancer as a target, you must configure Turbonomic to discover the virtual applications that load balancer manages.

The steps to do this are:

1. **Specify discovery for the applications the load balancer will manage**

   The load balancer manages workload across instances of running applications in your environment. Turbonomic must be configured to discover and manage these running applications before it can recognize that they are managed by the load balancer. For example, to specify discovery of a specific application, you define its application signature and then provide credentials for the protocol (WMI, SNMP, or JMX) that Turbonomic can use to access the VMs that host the application. For more information, see "Application Discovery" in the Turbonomic User Guide.

2. **Assign application signatures to specific virtual applications**

   Turbonomic discovers the virtual applications that the load balancer uses to manage applications. After you have specified signatures and discovery for the applications you want the load balancer to manage, you can then assign those application signatures to specific virtual applications. For more information, see "Load Balancer Discovery" in the Turbonomic User Guide.

   For example, assume the load balancer uses virtual applications named *web* and *sql* to manage web servers and SQL databases. Also assume that you have defined an application signature named IIS for IIS Web Server applications, and another signature named SQL for Microsoft SQL Server applications. In that case, you need to map the IIS signature to the web virtual application, and SQL to the sql virtual application (as explained in the Load Balancer Discovery section). Now Turbonomic can discover these applications and represent them as the running components within the load balancer.

Adding Citrix NetScalar Targets

To add NetScaler targets, select the **Load Balancers > NetScaler** option on the Target Configuration page and provide the following information:

- **Address:** The IP address of the primary NetScaler load balancer server

  Turbonomic discovers the service types that are assigned to that load balancer. These service types appear in the Turbonomic supply chain as virtual applications. To enable discovery of virtual applications, you must configure load balancer discovery.
- **Username/Password:** The credentials of an account Operations Manager can use to connect to the load balancer.

Actions

Turbonomic recommends actions for the Load Balancer supply chain as follows.

Entity Type	Action
Virtual Applications	Turbonomic does not recommend actions to perform on the virtual application itself, but it does recommend actions to perform on the VMs that host bound applications. For example, assume a virtual application that manages three SQL databases. If a surge in requests degrades performance across all three databases, then Turbonomic can start a new VM to run another instance of the database application, and bind it to the virtual application. On the other hand, if SQL requests drop off so that the load balancer only forwards requests to two of the databases, Turbonomic can suspend the dormant database and unbind it from the virtual application. Virtual Applications will only be discovered if there is a load balancer configured.
Applications	Without discovered Guest OS Processes, Turbonomic doesn’t perform actions on applications. Instead, it performs resize actions on the host VMs. If host utilization is high enough on the physical machine running the application VM, Turbonomic may also recommend provisioning a new host. For specific applications, see the individual application entry.
Virtual Machines	• Provision additional resources (VMem, VCPU) • Move Virtual Machine • Move Virtual Machine Storage • Reconfigure Storage • Reconfigure Virtual Machine

Monitored Resources

Turbonomic monitors the following resources for the Load Balancer supply chain:

Entity Type	Commodity
Virtual Applications	• Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second
Application	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB)

	• Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Transactions The utilization of the allocated transactions per second for the given virtual application Measured in transactions per second • Heap The utilization of the application server's heap Measured in Kilobytes (KB) • Response Time The utilization of the server's allocated response time Measured in Milliseconds (ms) • Threads The utilization of the server's thread capacity Measured in Threads • DBMem The utilization of the database's memory capacity. Only applicable to database servers Measured in Kilobytes (KB) • Connection The utilization of the connection capacity. Only applicable to database servers Measured in Connections • TransactionLog The utilization of the server's capacity for storage devoted to transaction logs Measured in Kilobytes (KB) • Garbage Collection The of server uptime spent garbage collecting Measured in of uptime (%)
VirtualMachine	• Virtual Memory (VMem) The utilization of the VMem allocated to the hosting VM Measured in Kilobytes (KB) • Virtual CPU (VCPU) The utilization of the VCPU allocated to the hosting VM Measured in Megahertz (MHz) • Virtual Storage (VStorage) The utilization of the virtual storage capacity allocated for the VM Measured in Kilobytes (KB) • Storage Access Operations Per Second (IOPS) The utilization of IOPS allocated for the VStorage on the VM Measured in IOPS • Latency The utilization of latency allocated for the VStorage on the VM Measured in milliseconds (ms)

Turbonomic Targets for Aggregation

For large virtual environments, you can use more than one Turbonomic instance to manage your workload. By aggregating multiple instances in this way, you can view the entire environment through a single GUI. One special installation of Turbonomic serves as the aggregating instance to display the combined data of each underlying target instance. For more information about Turbonomic aggregation, see "AggregatedInstTurbonomic allations" in the Turbonomic User Guide.

License Requirements

All Editions of Turbonomic support aggregated deployments.

Prerequisites

The Turbonomic instance that aggregates other Turbonomic servers cannot have any other targets — It can only manage other instances of Turbonomic.

Adding Turbonomic Targets

To add an aggregated instance of Turbonomic, select the **Turbonomic Appliance** option on the Target Configuration page and provide the following information:

NOTE: All the targets of an aggregating instance must be target instances of Turbonomic. You cannot use an aggregating instance to manage other types of targets.

- Address
  The name or IP address of the Turbonomic server.
- Username/Password
  Credentials for the administrator account on the Turbonomic server.
- Appliance ID
  An ID string that you can use to identify this particular Turbonomic server.

  The Server ID appears in the aggregating user interface when the user needs to access an underlying server. For example, assume you're logged into an aggregating server and want to run a plan. When you display the Plan view, you will see the underlying target servers, listed by their Server IDs.

Appendix — Target Configuration

This appendix contains topics that are related to configuring Turbonomic targets.

Enabling Management via DCOM

NOTE: This section applies to Operations Manager 5.3 or below only. If you are using Operations Manager 5.4 or higher, see Enabling Windows Remote Management on page 191.

To enable management via DCOM, the account you use to connect to each Hyper-V target must be one of the server's "WBEM Scripting Locator" owners. To enable access, edit the server’s WBEM Scripting Locator registry keys as described below to add the account and grant full control.

1. **Open regedit as Administrator.**
2. **Locate the following registry key:**
   `HKEY_CLASSES_ROOT\CLSID\{76A64158-CB41-11D1-8B02-00600806D9B6}`
3. **Right-click the key and choose Permissions.**
4. **Click Advanced** and then click the **Owner tab.**
5. **In the Owners list, add the user you want to allow to connect to the machine.**
6. **Click OK.**
7. **Highlight the user and grant Full Control.**
8. **Search for the following registry key:**
   `HKLM\Software\Classes\Wow6432Node\CLSID\{76A64158-CB41-11D1-8B02-00600806D9B6}`
9. **If the key exists, repeat the previous steps to grant full control for this key also.**

NOTE: The user created or modified here must also be the user whose credentials are given for target addition in the Username and Password fields of "Adding Hyper-V Targets" in the Turbonomic User Guide

Configuring your Firewall for DCOM

Your firewall must be configured to enable communication over DCOM from Operations Manager to the Hyper-V server. DCOM uses port 135 to make the initial connection, but port 445 must also be open.

After the initial connection, DCOM switches to a random non-privileged port for ongoing communication. Most firewalls detect this switch and enable communication via the new port automatically. However, Microsoft's own default firewall setup does not support automatic port switching. If you're using the default Microsoft firewall, you have two options:

- Configure each Hyper-V target to use a static port for DCOM. For steps to do this, see https://msdn.microsoft.com/en-us/library/windows/desktop/bb219447(v=vs.85).aspx. After configuring the static port on each Hyper-V server, open the port on your firewall to allow communication from Operations Manager to your Hyper-V servers.
- Configure the firewall to enable port switching using the following command:

```
netsh advfirewall firewall add rule name = Turbonomic dir = in protocol = tcp
action = allow localport = rpc remoteip = <IP address of Turbonomic
appliance> profile = DOMAIN
```

If you encounter problems, the following article provides troubleshooting guidance: http://blogs.technet.com/b/askperf/archive/2007/06/22/basic-wmi-testing.aspx.

Secure Setup of WSMan

Turbonomic provides a secure option for Hyper-V/VMM Targets which requires that WSMan be set up securely. Use PowerShell to generate a self-signed certificate, and create an HTTPS WinRM listener.

NOTE: For clustered Hyper-V targets, you do not need to create a listener on each host. Only create a listener on the host that is being added to the "Address" field in the Target Configuration.

To enable secure WSMan on your Hyper-V host:

1. **Generate a self-signed certificate using the following command:**

```
New-SelfSignedCertificate -CertstoreLocation Cert:\LocalMachine\My -DnsName
"myhost.example.org"
```

2. **Find the thumbprint for the certificate for the host:**

```
Get-childItem cert:\LocalMachine\My
```

3. **Create an HTTPS WinRM listener for the host with the thumbprint you've found:**

```
winrm create winrm/config/Listener?Address=*+Transport=HTTPS
'@{Hostname="myhost.example.org";
CertificateThumbprint="THUMBPRINT_YOU_FOUND"}'
```

4. **Verify the presence of configured listeners:**

```
Get-WSManInstance –ResourceURI winrm/config/listener –Enumerate
```

Enabling Windows Remote Management

NOTE: This topic covers the setup of WinRM for a single host. For enabling WinRM on multiple hosts, see the following topics:

- Enabling WinRM Via Global Policy Objects on page 191
- Enabling WinRM Via PowerShell on page 193

Turbonomic communicates with your Hyper-V servers using Web Services Management (WS-Management), which is implemented on Microsoft platforms using Windows Remote Management (WinRM). The following steps show how to enable WinRM on a single host, using the command line.

1. **Ensure Windows Firewall is running on the host.**

   For you to configure WinRM successfully, Windows Firewall must be running on the host. For more information, see the Microsoft Knowledge Base article #2004640 (http://support.microsoft.com/kb/2004640).

2. **Set up a Service Principal Name (SPN) for the host machine.**

   The machine must have a SPN of the form, `protocol/host_address`. For example, `WSMAN/10.99.9.2`.

   To get a list of SPNs for the machine, execute the following in the command window:

   ```
   setspn -l my-vmm-server-name
   ```

   If there is no valid SPN in the list, create one by running the command:

   `setspn -A protocol/host-address:port` where `port` is optional

   For example, `setspn -A WSMAN/10.99.9.2:VMM-02`

3. **Set up the Windows Remote Management (WinRM) service to run on startup.**

   Run the `quickconfig` utility to set up the WinRM service. The `quickconfig` utility:

   - Configures the WinRM service to auto-start
   - Configures basic authentication and disables unencrypted traffic
   - Creates a firewall exception for the current user profile
   - Configures a listener for HTTP and HTTPS on any IP address
   - Enables remote shell access

   To run `quickconfig`, log into a command window as Administrator on the host machine. Then execute the following commands:

   ```
   winrm quickconfig
   ```

   Enter `y` to accept the `quickconfig` changes

4. **Set permissions on the host machine.**

   Execute the following commands in the command window to modify the settings made by `quickconfig`:

   - To set the memory capacity for remote shells:

     ```
     winrm set winrm/config/winrs @{MaxMemoryPerShellMB="1024"}
     ```

   - To set up an unsecured HTTP connection:

     ```
     winrm set winrm/config/service @{AllowUnencrypted="true"}
     winrm set winrm/config/service/Auth @{Basic="true"}
     ```

Enabling WinRM Via Global Policy Objects

You can configure WinRM for all of your Hyper-V targets by creating and linking a Global Policy Object (GPO) within the Hyper-V domain and applying the GPO to all servers.

Follow the steps below to enable Windows Remote Management (WinRM) for your Hyper-V targets.

For additional information, you can also see the Green Circle article, Application Discovery: GPO for necessary Registry Changes

1. **On the AD domain controller, open the Group Policy Management Console (GPMC). If the GPMC is not installed, see** https://technet.microsoft.com/en-us/library/cc725932.aspx**.**
2. **Create a new Global Policy Object:**
   a) In the GPMC tree, right-click **Group Policy Objects** within the domain containing your Hyper-V servers.
   b) Choose **Create a GPO in this domain,** and link it here.
   c) Enter a name for the new GPO and click **OK**.
3. **Specify the computers that need access:**
   a) Select the new GPO from the tree.
   b) On the **Scope** tab, under **Security Filtering**, specify the computer or group of computers you want to grant access. Make sure you include all of your Hyper-V targets.
4. **Right-click the new GPO and choose Edit to open the Group Policy Management Editor.**
5. **Configure the WinRM Service:**
   a) In the Group Policy Management Editor, select **Computer Configuration > Policies > Administrative Templates > Windows Components > Windows Remote Management (WinRM) > WinRM Service**.
   b) Double-click each of following settings and configure as specified:

Allow automatic configuration of listeners ("Allow remote server management through WinRM" on older versions of Windows Server):	Enabled IPv4 filter: *
Allow Basic authentication:	Enabled
Allow unencrypted traffic:	Enabled

6. **Configure the WinRM service to run automatically:**
   a) In the Group Policy Management Editor, expand **Computer Configuration > Preferences > Control Panel Settings**.
   b) Under Control Panel Settings, right-click Services and choose **New > Service**.
   c) In the New Service Properties window, configure the following settings:

Startup:	Automatic
Service name:	WinRM
Service option:	Service start

7. **Enable Windows Remote Shell:**
   a) In the Group Policy Management Editor, select **Computer Configuration > Policies > Administrative Templates > Windows Components > Windows Remote Shell**.
   b) Double-click the following setting and configure as specified:

Allow Remote Shell Access:	Enabled

8. **Add a Windows Firewall exception:**
    a) In the Group Policy Management Editor, open **Computer Configuration > Windows Settings > Security Settings > Windows Firewall > Windows Firewall**.
    b) Under Windows Firewall, right-click **Inbound Rules** and choose **New > Rule**.
    c) In the New Inbound Rule Wizard, select **Predefined: Windows Remote Management and Allow the connection**.

The new group policy will be applied during the next policy process update. To apply the new policy immediately, execute the following command at a Powershell prompt:

```
gpupdate /force
```

Enabling WinRM Via PowerShell

Using PsExec, you can run quickconfig on all your Hyper-V servers and change the default settings remotely. PsExec is a component of PsTools, which you can download from https://technet.microsoft.com/en-us/sysinternals/bb897553.aspx.

1. **Create a text file containing the Hyper-V host names, for example:**

```
hp-vx485
hp-vx486
```

2. **Since Turbonomic requires changes to the default quickconfig settings, create a batch file containing the following command:**

```
@echo off Powershell.exe Set-WSManQuickConfig -Force Powershell.exe Set-Item
WSMan:\localhost\Shell\MaxMemoryPerShellMB 1024
```

NOTE: If you are connecting via HTTP, you must include the following command:

```
Powershell.exe Set-Item WSMan:\localhost\Service\AllowUnencrypted -Value
    $True
```

3. **Use PsExec to enable WinRM on the remote servers:**

```
.\PsExec.exe @<hosts_file_path> -u <username> -p <password> -c
<batch_file_path>
```

NOTE: If you get an error message when executing this command, add the `-h` option (`.\PsExec.exe -h`).

Port Configuration

To support communication with different target technologies, Turbonomic requires open access to the following ports:

Target	Port(s)
Vcenter (Monitoring)	80, 443
Vcenter (Tags)	443, 10443
Hyper-V	5985, 5986
RHEV	80, 443
XenCenter	80, 443
Vcloud Director	80, 443
CloudStack	80, 443
OpenStack	5000, 8774, 8776, 8777
VMM	5985, 5986
CloudFoundry	80, 443
VMAX	5988, 5989
Dell Compellent	5988, 5989
Nutanix	9440
XtremIO	80, 443
Pure	80, 443
HP 3PAR	5988, 5989, 8080

NetApp	80, 443
NetScaler	80, 443
Cisco UCS	80, 443
sFlow	80, 443
Netflow	80, 443
Arista	80, 443
WebSphere	8880
Weblogic	7001
Exchange	5985, 5986
Tomcat	1009
JBOSS	9990
Oracle	1521
SQL Server	1433
MySQL	3306

Cisco Unified Computing System

UCS Blade Provisioning

When managing a UCS fabric target, Turbonomic can provision any blade servers that are installed in a chassis but not currently in operation. If the workload demands more physical compute resources, Turbonomic can automatically direct UCS to provision a blade, or it can recommend that you provision a blade and you can execute the action from the To Do list. To enable this capability, you must perform two basic steps:

- Configure the way UCS and vCenter Server manage information as blades are provisioned

  To enable Turbonomic to perform automatic provisioning of UCS blades, you must configure UCS to work with vCenter Server as they both manage resources such as server pools, policies, and Service Profile Templates. This ensures that as Turbonomic directs the UCS Manager to provision a new blade, vCenter Server will recognize that the new physical host is available. Once vCenter Server can recognize the new blade, Turbonomic can direct vCenter Server to move workloads onto the server.

  Turbonomic provisions new blades based on the service profiles of operating blades. To enable this, the configuration must include Service Profile Templates, and the operating blades must be bound to these templates.

  For information about configuration that enables automated provisioning of blades, see the Cisco Communities post, “UCS PowerTool and VMware PowerCLI automated management of Auto-deploy” at the following location:

  `https://communities.cisco.com/community/technology/datacenter/ucs_management/cisco_ucs_developed_integrations/blog/2013/09/16/ucs-powertool-and-vmware-powercli-automated-management-of-auto-deploy`

  This post includes a video that shows “a joint PowerShell integration utilizing both Cisco UCS PowerTool and VMware PowerCLI.” You can also download the scripts from this post and modify them as necessary for your deployment.

- Set the Host Provision action to Automate or Manual for the blade servers. By default, Turbonomic sets the Host Provision action to Recommend.

  For any hosts other than blade servers managed by UCS, Turbonomic cannot provision hosts automatically or manually. Instead, it recommends that you provision a host, and you then install the physical machine and wire it to the network.

  In a UCS deployment you can have blade servers installed in the chassis and ready to provision. In that case, Turbonomic can direct UCS to provision a new blade to meet workload demands. For these servers, you can set the Host Provision action to Automated or Manual.

NOTE: It's important that you only set Automated or Manual host provisioning to UCS blades. If you set Host Provision to Automated for other types of hosts, Turbonomic will attempt to perform the action and fail. As a result, you might never see the recommendation to provision a new host of that type.

Turbonomic groups blade servers by chassis. To restrict Automated or Manual settings to blade servers, use this group. You can set the action mode for all blade servers in your environment, or you can set the mode differently for individual chassis.