

Organizing Your Environment

Dynatrace Training Module



Agenda

- Host Groups
- Network Zones
- Metadata and Tagging
- Naming Rules
 - Host
 - Process Group
 - Service
- Management Zones

Host Groups

Why would I need a Host Group?

- Host Groups affect how process groups are detected. By default, similar processes and services are detected and grouped together.
- Host Groups serve as a method of automatically separating processes and services by the group name
 - An example is "Environment_AppName_Function"
 - Such as "Prod_EasyTravel_Webservers" vs. "Dev_EasyTravel_WebServers"
 - A Host Group will be the same on hosts that are performing the same role.
- Use Cases:
 - Monitoring multiple application environments (DEV, INT, PRD) in the same Dynatrace environment or tenant.
 - Need to split process groups automatically, without defining environment variables per host/process
 - Host Groups can be used as filters and for defining Auto-Tagging Rules and Management Zones.

<https://www.dynatrace.com/support/help/shortlink/host-groups>

Host Groups

Deployment status

OneAgents

Deployment status

OneAgents

ActiveGates

Network zones

OneAgent deployment

Manage your hosts and agents

Add new host

Filter by

Show only recently connected hosts (last 72h)

Showing 3 hosts.

Host unit consumption: 1 (for 3 total hosts)

OS	Host name	Version	Global monitoring	Details
	<div>Dev_EasyTravel – ETSvr-2 – Windows Server 2012 R2 Standard</div> <div>Host units: 0.5 Host group: Dev_EasyTravel IP addresses: 10.0.2.5, 192.168.192.77, 192.168.1.18</div>	1.217.149		
	<div>Prod_EasyTravel – ETSvr – Windows Server 2012 R2 Standard</div> <div>Host units: 0.5 Host group: Prod_EasyTravel IP addresses: 192.168.192.76, 10.0.2.4</div>	1.217.149		

4 Groups

Name

Dev_EasyTravel - IIS app pool dotNetBackend_easyTravel_x64

Dev_EasyTravel - IIS app pool dotNetFrontend_easyTravel_x64

Prod_EasyTravel - IIS app pool dotNetBackend_easyTravel_x64

Prod_EasyTravel - IIS app pool dotNetFrontend_easyTravel_x64

5 Services

Name

ASP.NET

Dev_EasyTravel – dotNetFrontend_easyTravel_x64:9000

Dev_EasyTravel – IIS app pool dotNetFrontend_easyTravel_x64

Dev_EasyTravel – easyTravel Customer Frontend

Dev_EasyTravel – easyTravel (x*)

ASP.NET

Prod_EasyTravel – dotNetFrontend_easyTravel_x64:9000

Prod_EasyTravel – IIS app pool dotNetFrontend_easyTravel_x64

Prod_EasyTravel – easyTravel Customer Frontend

Prod_EasyTravel – easyTravel (x*)

ASP.NET

Prod_EasyTravel – IIS app pool dotNetFrontend_easyTravel_x64 on port 9000

Prod_EasyTravel – IIS app pool dotNetFrontend_easyTravel_x64

How to Set or Adjust the Host Group

- This option should be set during install as changes or updates will require server access and application process restarts
- A Host Group defined on a host during OneAgent installation
 - AIX, Linux, and Windows installers allow you to set installation option for that on the command-line
 - `--set-host-group=env_application_function`
 - The Windows installer in GUI mode allows for additional configuration options
 - MSI installation parameters are available as well:
<https://www.dynatrace.com/support/help/shortlink/windows-custom-installation#msi-installation-parameters>
 - After the installation is complete the Host Group is statically assigned to the host
- A Command Line Interface (CLI) "oneagentctl" utility allows setting or updating the Host Group
 - The OneAgent will need to be shutdown before, and restarted after, setting the host group.
 - Add `--restart-service` to the CLI to automate the restart
- Each host belongs to at most one host group

How to Set the Host Group

- Utilize the OneAgent customization settings when retrieving the software
- Filing in the Host Group will add the necessary parameter to the command line in step 4.

3. Set customized options (optional).

✓ Set Host Groups, Network Zones and tags.

Network zone	default	What are network zones?
Host group	Prod_EasyTravel	What are host groups?
Custom host name		What are custom host names?
Tags	Prod_DT-SaaS	What are tags?
Properties	Dev_EasyTravel	What are properties?

☒ Enable full-stack monitoring of infrastructure, application performance and user experience data.
[What's included in full-stack monitoring?](#) If full-stack monitoring is disabled, OneAgent operates in infrastructure-only monitoring mode.

☒ Enable access to application log-file content on this host for problem analysis.
[What insights can Log Monitoring provide?](#)

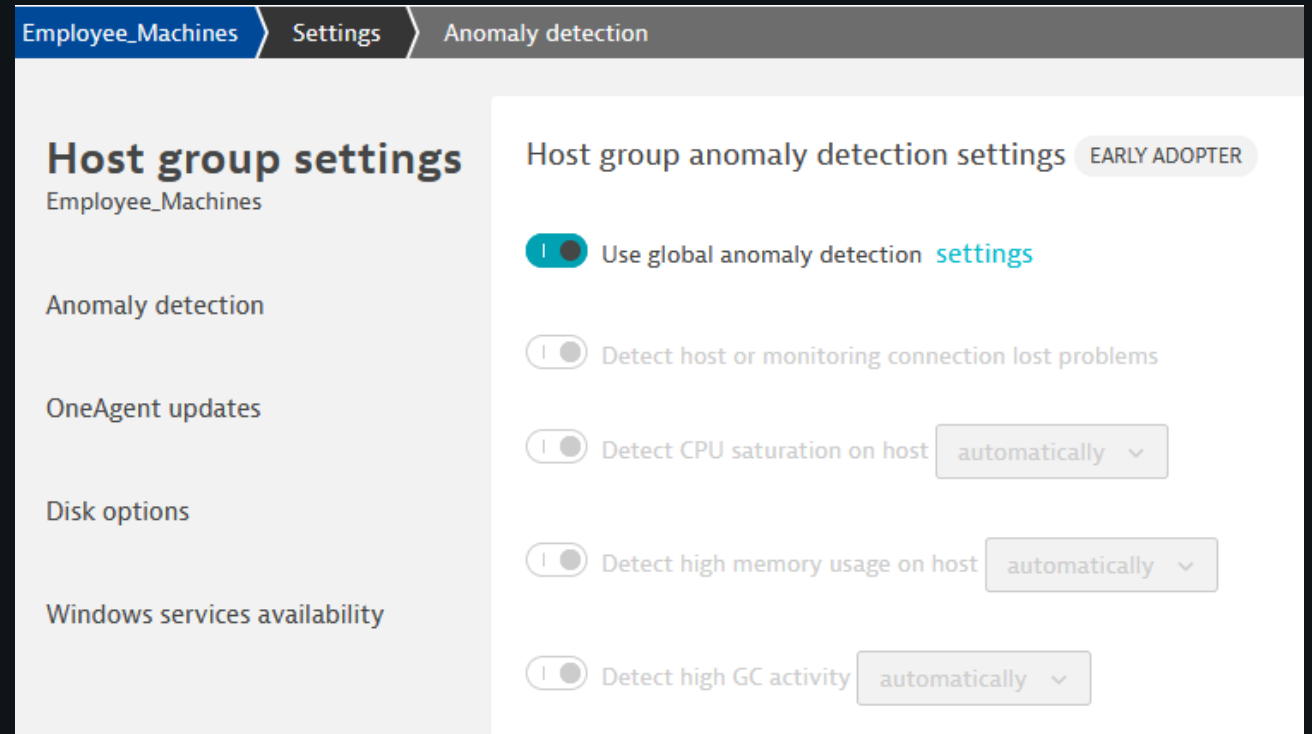
4. Run the installer with root rights.

```
/bin/sh Dynatrace-OneAgent-Linux-1.217.149.sh --set-host-group=Prod_EasyTravel
```

Copy

How else does it affect the monitoring?

- Hosts can be configured on the host group level
- This makes changing settings for a set of hosts much easier
- Settings that can be defined per Host Group:
 - Alerting thresholds
 - OneAgent updates
 - Disk monitoring options
 - Windows services monitoring



Host Group Uses

- Host groups can also be used in tagging rules and for defining Management Zones so you can apply additional context information to the different entities in Dynatrace based on host groups

Tag name

Add a new rule

Filter by rule name

Rule

Process groups get value '{HostGroup:Name/^[^_]+})' where Kubernetes namespace does not exist and that run on **Hosts** where Host group name contains regex '^[^_]+')

Management zone name

Add a new rule

Filter by rule name

Rule

Process groups on **Hosts** where Host group name begins with 'Prod'

Network Zones

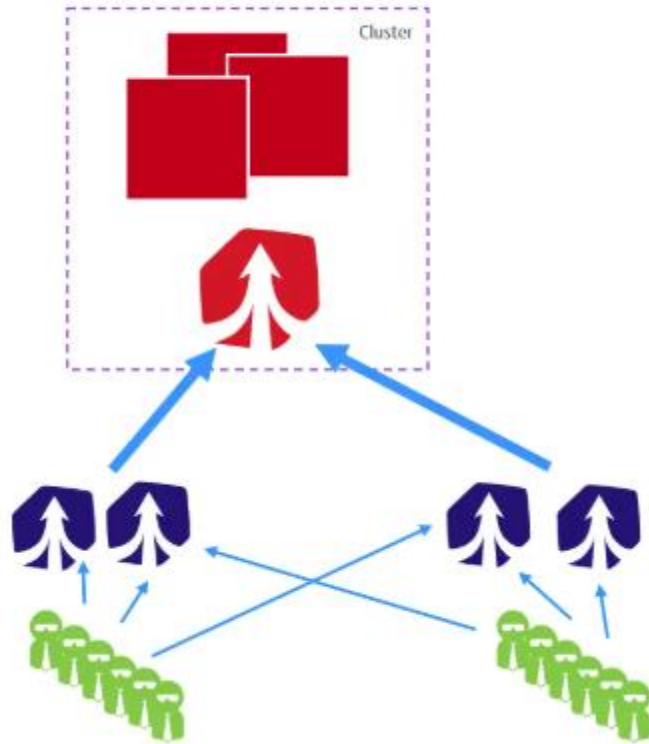
Why would I need a Network Zone?

- Network zones are Dynatrace entities that represent your network structure.
- They help route OneAgent traffic efficiently, avoiding unnecessary traffic across data centers and network regions.
- Without Network Zones OneAgent traffic is load balanced among all ActiveGates, regardless of location
- Network Zone names should follow the network structure:
https://www.dynatrace.com/support/help/shortlink/network-zones-basics#anchor_naming
- Alternative network zones can be configured in the UI to serve as backup zones
- OneAgents are still aware of all ActiveGate and Cluster endpoints and will automatically failover if necessary
- Use Cases
 - Optimize traffic flow from OneAgents to ActiveGates
 - Eliminate unnecessary cross-datacenter traffic
 - Take full advantage of the compression offered by ActiveGates

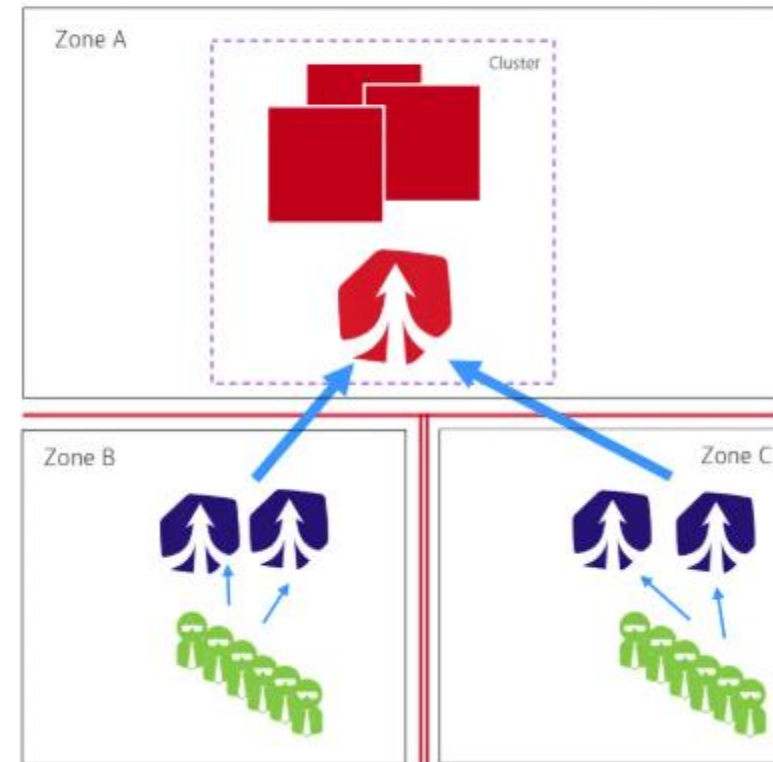
<https://www.dynatrace.com/support/help/shortlink/network-zones>

Why would I need a Network Zone?

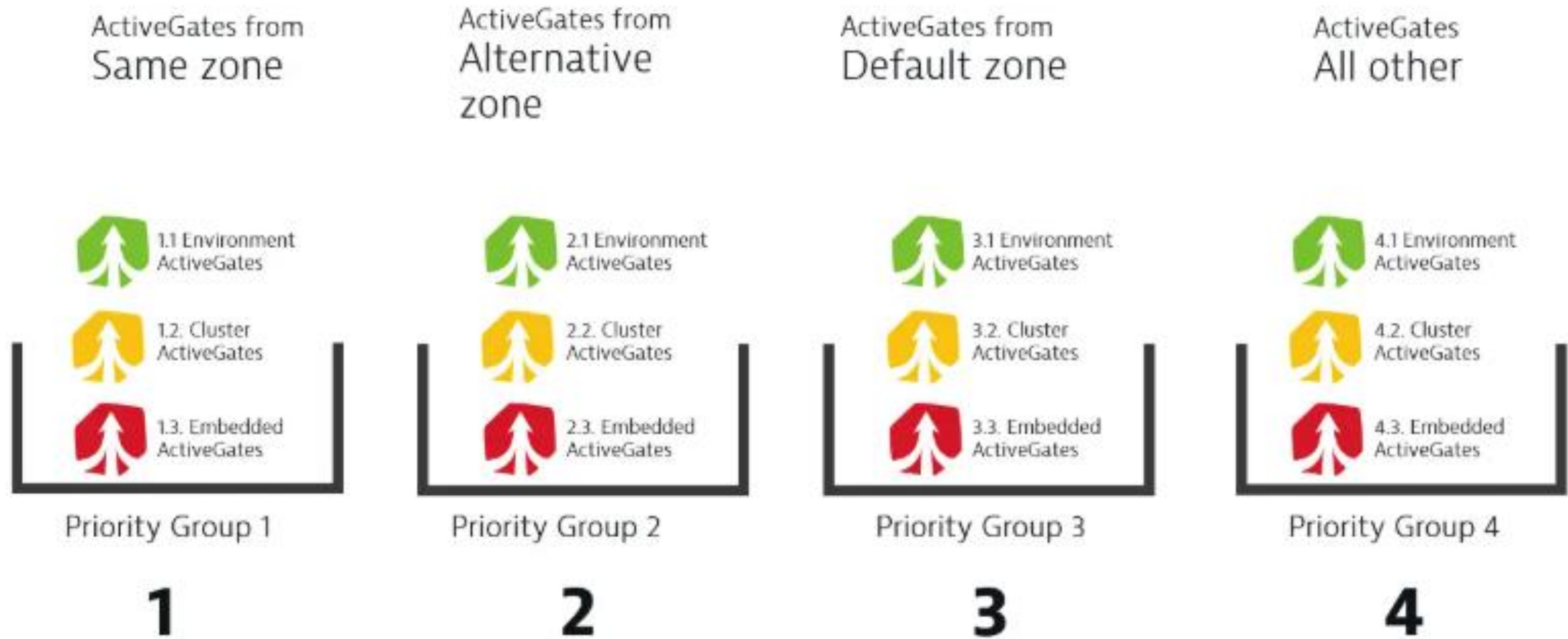
Without network zones



With network zones



Network Zones – Connection Priority



How to Set or Adjust the Network Zone for an ActiveGate

- This option should be set during ActiveGate installation. Changes or updates will require server access and an ActiveGate service restart.
- An ActiveGate can be assigned to a network zone at the time of installation
 - Installers allow you to set the Network Zone on the command-line
 - `--set-network-zone=My.Network.Zone`
- Alternatively, the ActiveGate configuration can be changed after installation.
 - Modify the "custom.properties" file in the ActiveGate configuration directory:
<https://www.dynatrace.com/support/help/shortlink/sgw-files#directory-locations>
 - Add the necessary settings: <https://www.dynatrace.com/support/help/shortlink/sgw-configure#section-connectivity->
 - Restart the ActiveGate Service: <https://www.dynatrace.com/support/help/shortlink/stop-restart-activegate>

How to Set or Adjust the Network Zone for a OneAgent

- This option should be set during OneAgent installation. Changes or updates will require server access and a OneAgent service restart. Application processes will not need to be restarted.
- A Network Zone can be defined on a host during OneAgent installation
 - AIX, Linux, and Windows installers allow you to set installation option for that on the command-line
 - `--set-network-zone=My.Network.Zone`
 - The Windows installer in GUI mode allows for additional configuration options
 - MSI installation parameters are available as well:
<https://www.dynatrace.com/support/help/shortlink/windows-custom-installation#msi-installation-parameters>
 - After the installation is complete the OneAgent will attempt to communicate with the ActiveGates in the same zone
- A Command Line Interface (CLI) "oneagentctl" utility allows setting or updating the Network Zone
 - The OneAgent will need to be shutdown before, and restarted after, setting the host group.
 - Add `--restart-service` to the CLI to automate the restart

How to Set the Network Zone

- Utilize the OneAgent customization settings when retrieving the software
- Filing in the Network Zone will add the necessary parameter to the command line in step 4.

3. Set customized options (optional).

✓ Set Host Groups, Network Zones and tags.

Network zone

et.fl.vmware.linux

Host group

default

Custom host name

et.fl.vmware.linux

Tags

Properties

[What are network zones?](#)

[What are host groups?](#)

[What are custom host names?](#)

[What are tags?](#)

[What are properties?](#)



Enable full-stack monitoring of infrastructure, application performance and user experience data.

[What's included in full-stack monitoring?](#) If full-stack monitoring is disabled, OneAgent operates in infrastructure-only monitoring mode.



Enable access to application log-file content on this host for problem analysis.

[What insights can Log Monitoring provide?](#)

4. Run the installer with root rights.

```
/bin/sh Dynatrace-OneAgent-Linux-1.217.149.sh --set-network-zone=et.fl.vmware.linux
```

Copy

Network Zones Status

Manage

Dynatrace Hub

Deploy Dynatrace

Deployment status

Settings

Deployment status > Network zones

Deployment status

OneAgents

ActiveGates

Network zones

Network zones Early adopter

Network zones allow you to effectively manage the Dynatrace monitoring environment. Thanks to them, you can group ActiveGates in any way adopted to the needs of the monitored environment.

Search...

Network zone	OneAgents	ActiveGates
default	0	0
et.fl.vmware.linux	3	1
et.fl.vmware.windows	1	1

Network Zones Status and Alternate Network Zones

Deployment status

Network zones

et.fl.vmware.linux

Deployment status

OneAgents

ActiveGates

Network zones

et.fl.vmware.linux

Early adopter

Edit

ActiveGates

1
all ActiveGates

OneAgents

3
all OneAgents

3
OneAgents configured in Network Zone

0
OneAgents using this zone as alternative

Alternative network zones

By adding an alternative network zones, you can specify how routing will occur if this network zone is unavailable.

Search...

Alternative network zone	OneAgents	ActiveGates
et.fl.vmware.windows	1	1

Metadata and Tags

Why would I need Metadata and Tags

- Metadata and Tags enable you to organize your monitored environments in a meaningful way.
- Metadata are key/value pairs that are detected, or added, on a monitored entity.
- Metadata are mainly used for defining extra information for entities
- Tags are typically based on detected, or added, metadata.
- Tags in Dynatrace are key/value labels or markers
- Tags are used for filtering or organizing entities
- When creating tags be sure to review the metadata detected in the properties of an entity
- Although tags and metadata are closely related, they are different concepts and are created and used in different ways
- <https://www.dynatrace.com/support/help/shortlink/tags-and-metadata-hub>

Tags and Metadata

Hosts Prod_EasyTravel - ETSvr - Windows Server 2012 R2 Standard

Prod_EasyTravel - ETSvr - Windows Server 201...
Uptime: 4 minutes

✓ Properties and tags

Tags

Application: EasyTravel CPU Cores: 2 DB Vendor: MySQL Client DB Vendor: SQL Server CE Environment: Prod
HCPOwner: [redacted] Host Group: Prod_EasyTravel IIS App Pool: dotNetBackend_easyTravel_x64
IIS App Pool: dotNetFrontend_easyTravel_x64 LOB: Travel OS: Windows
OS Info: Windows Server 2012 R2 Standard Priority: Critical VM Servers: [redacted] + Add tag

Windows Server 2012 R2 Standard, ver. 6.3.9600

Detected Metadata

Detected name: ETSvr
OneAgent version: 1.217.149.20210528-111116
Architecture: x86, 64-bit
Data center: United States
Host group: Prod_EasyTravel
IP addresses: [redacted]
Logical CPU cores: 2
Management zones: Operations Team, App:Easytravel, Env:Prod, and Prod
Monitoring mode: Full stack
Physical CPU cores: 2
Virtualization: VirtualBox

Added Metadata

Environment custom meta data
Business_Owner: Travel_Department
HCPOwner: [redacted]
Priority: Critical
Support_Team: IT_Operations
[redacted]: [redacted]

Tags can be used to filter Views, Charts and Alerts

Filtered by Tag:

Application
CPU Cores
Environment
HCPOwner
Host Group
LOB
OS
OS Info
Owner
Priority
VM Servers

Chart type: Line

Dimensions: Host

Chart results

Filtered by

Error alert (After 0 mins; Only include entities that have any tags Application:EasyTravel)

Slowdown alert (After 30 mins; Only include entities that have any tags Application:EasyTravel)

Best Practices and recommendations for tagging

- Typically, you should think about using standardized or additional metadata and not about tags
- Be sure to view the properties of an entity for available metadata
- Add metadata that helps to clarify, manage or organize the environment data
- Added metadata should be standardized across the environment as much as possible
- Areas to consider
 - Owner/team/business unit/line of business
 - Environment: staging/production
 - Importance/severity (relevant for alerting profiles)
- Assign metadata and create tags in key-value pairs (e.g. LOB=Travel not just "Travel")
- Add Host metadata for host information common to all processes on the host
- Process metadata can be added for process-specific information
- Auto-tagging based on metadata allows tags to be generated automatically and assigned to monitored entities

<https://www.dynatrace.com/support/help/shortlink/tagging-best-practices>

How to Set or Adjust Host Metadata

- Host metadata can be set during OneAgent installation or adjusted afterward. Changes or updates will require server access. No services need to be restarted.
 - Spaces cannot be used in a key or value
 - Multiple properties can be set at the same time
- Host metadata can be defined during OneAgent installation
 - AIX, Linux, and Windows installers allow you to set installation options on the command-line
 - `--set-host-property=key=value`
 - `--set-host-property=LOB=Travel --set-host-property=priority=tier-1`
 - The Windows installer in GUI mode allows for additional configuration options
- A Command Line Interface (CLI) "oneagentctl" utility allows setting or updating the host properties
 - Utilize the same parameters as during OneAgent installation
 - `--set-host-property=key=value`
 - `--set-host-property=LOB=Travel --set-host-property=priority=tier-1`

<https://www.dynatrace.com/support/help/shortlink/oneagentctl#custom-host-metadata>

How to Set Metadata

- Utilize the OneAgent customization settings when retrieving the software
- Filing in the Properties will add the necessary parameter to the command line in step 4.

3. Set customized options (optional).

✓ Set Host Groups, Network Zones and tags.

Network zone	<input type="text" value="default"/>	What are network zones?
Host group	<input type="text"/>	What are host groups?
Custom host name	<input type="text"/>	What are custom host names?
Tags	<input type="text"/>	What are tags?
Properties	<div>LOB= Travel ✕ Priority= Tier1 ✕</div>	What are properties?

☒ Enable full-stack monitoring of infrastructure, application performance and user experience data.
[What's included in full-stack monitoring?](#) If full-stack monitoring is disabled, OneAgent operates in infrastructure-only monitoring mode.

☒ Enable access to application log-file content on this host for problem analysis.
[What insights can Log Monitoring provide?](#)

4. Run the installer with root rights.

```
/bin/sh Dynatrace-OneAgent-Linux-1.217.149.sh --set-host-property=LOB=Travel --set-host-property=Priority=Tier1
```

Copy

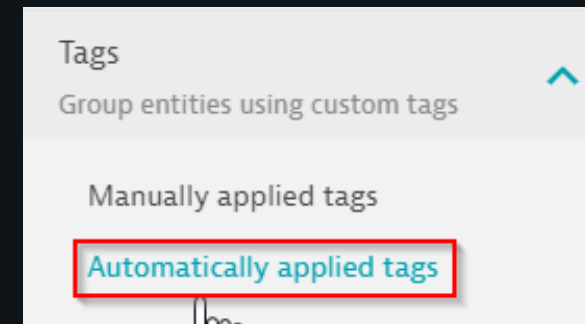
How to Set or Adjust Process Metadata

- Process metadata is set in technology-specific configuration files or via command line launch parameters. Changes or updates will require restarting the process.
 - Spaces cannot be used in a key or value
 - Multiple properties can be set at the same time
- Adding process metadata relies on adding a process environment variable
 - `DT_CUSTOM_PROP="key1=value1 key2=value2"`
 - `DT_CUSTOM_PROP="LOB=Travel Priority=Tier1"`
- Example:
 - Apache HTTP Web Server
 - Location of file: `/etc/sysconfig/httpd`
 - Add: `DT_CUSTOM_PROP="LOB=Travel Priority=Tier1"`

https://www.dynatrace.com/support/help/shortlink/process-group-properties#anchor_variables

Tagging Rules

- Types of Tags
 - Automatically based on Rules
 - Auto tagging rules can be changed and adapted any time and will apply almost immediately without any change to the monitored environment or applications.
 - Most commonly used
 - Most flexible
 - Dynamically added based on detected or standardized properties
 - Cannot be accidentally removed – as in manual tags
 - Manually via the UI or API
 - Can be difficult to manage in large environments
 - Must remember to add the tags
 - Easy to accidentally remove tags
 - Automatically without rules
 - Not very flexible
 - Useful for importing information from some platforms (Cloud Foundry, Kubernetes and OpenShift, AWS)



<https://www.dynatrace.com/support/help/shortlink/tagging>

Automatic Tags with Rules

Tagging - Automatically

Start by selecting Settings > Tags > Automatically applied tags

Settings > Tags > Automatically applied tags

Settings

- Monitoring
 - Setup and overview
- Processes and containers
 - Detection and naming
- Web and mobile monitoring
 - Real user and synthetic monitoring
- Cloud and virtualization
 - Connect cloud and virtualization types
- Server-side service monitoring
 - Manage and customize service monitoring
- Log Monitoring
 - Set up management of logs
- Anomaly detection
 - Configure detection sensitivity
- Alerting
 - Configure alerting settings
- Dashboards
 - Configure dashboard settings
- Integration
 - Integrate Dynatrace with 3rd party systems
- Topology model
 - Configure a domain-specific topology model
- Tags
 - Group entities using custom tags
 - Manually applied tags
 - Automatically applied tags**

Automatically applied tags

Tags simplify searches for related services, process groups, and hosts. They also facilitate the collection of related metrics into meaningful groups for analysis. [More...](#)

Create tag

Filter your tags by name

Name ▲	Delete	Edit
Application	×	✎
Business Owner	×	✎
CPU Cores	×	✎
DB Vendor	×	✎
DCPPriority	×	✎
Environment	×	✎
HCPOwner	×	✎
Host Group	×	✎
Host Support	×	✎
Hostname	×	✎

< 1 2 3 >

Either create a new tag or select a pre-existing tag

Edit a Tag – Creating a Rule

Edit tag

This custom tag is applied to all services, process groups, or hosts that match one or more of the rules defined below. [More...](#)

Tag name

Business Owner

Add a rule

Add a new rule

Filter by rule name

Rule	Delete	Disable/Enable	Edit
Process groups get value '{Host:Environment:Business_Owner}}' on Hosts where Business_Owner (Environment) exists	X	<input checked="" type="checkbox"/>	✓

Edit a Tag – Creating a Rule

Edit tag

This custom tag is applied to all services, process groups, or hosts that match one or more of the rules defined below. [More...](#)

Tag name

Business Owner

Add a new rule

Services

Optional tag value

Tag value {Service:Port}

Rule applies to Services of the following process group, service type, technology, and topology

Rule applies to entities matching the following properties

All process groups

All host groups

All service types

All technologies

All service topologies

Conditions

Akka actor system

begins with

☒ Case sensitive

Type value to be evaluated by rule

Remove condition

Add condition

☐ Apply to underlying hosts of matching services

☐ Apply to underlying process groups of matching services

Preview

Create rule

Cancel

Edit a Tag – Identify entities

Provide an optional tag value

Click and start typing to see the list of all available placeholders

You may also use a regex when using a placeholder {placeholder/regex}

Optional tag value

{Host:Environment}

{Host:Environment:Business_Owner}

{Host:Environment:HCPOwner}

{Host:Environment:HostSupport}

{Host:Environment:Meta-Data-Type}

{Host:Environment:Owner}

{Host:Environment:Priority}

Business_Owner (Environment) exists

Remove condition

Select which type of entity you are tagging

Optional tag value

Tag value {ProcessGroup:BinPath}

Rule applies to Process groups of the following technology type

Rule applies to entities matching the following properties

All host groups

All technologies

Conditions

Narrow the tag to a particular set of entities

NOTE:

Regular expressions are a powerful way to extract a certain pattern in a string.

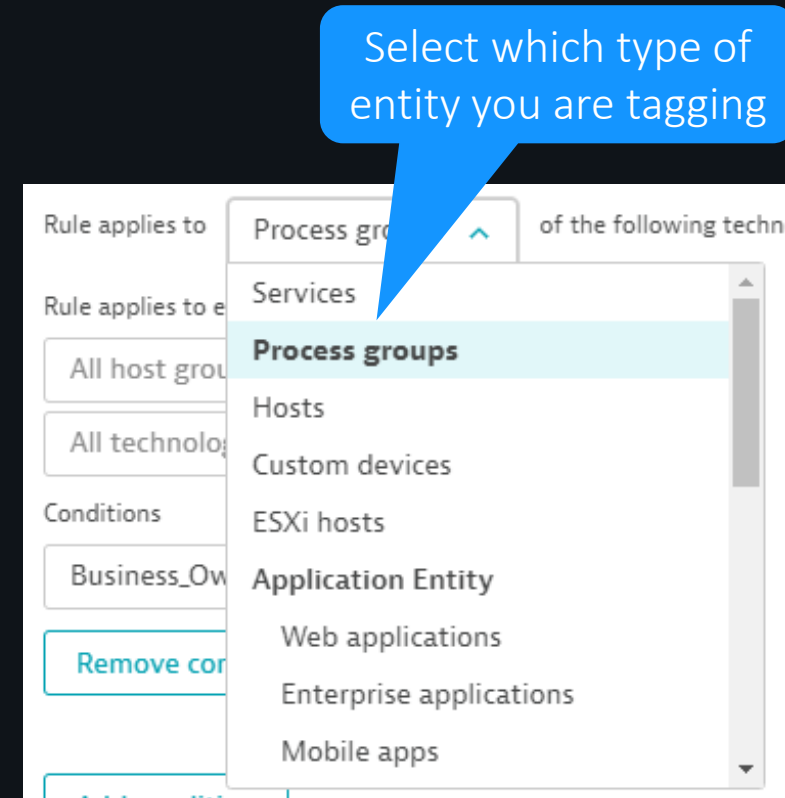
Regular expressions may also quickly become resource intensive.

Try to create efficient Regexp by avoiding lazy and greedy matching, backtracking, and repeated capture groups

Refer to: <https://www.dynatrace.com/support/help/shortlink/regex>

Applying rules to Process Groups vs. Services

- Dynatrace "Services" represent information about requests like web requests, web service calls, and queries
- Dynatrace "Process Groups" are a logical cluster of processes that are performing the same function across hosts
- Services only exist when requests are being sent.
- Process Groups exist as soon as the processes are started on a host.
- *Most of the time Process Groups should be tagged as the entity will exist as soon as a host/process is started.*
- Service entities do not exist until application requests have been sent.



Edit a Tag – Define Conditions

Conditions

Business_Owner (Environment)

.NET Core application DLL

☒ Case sensitive

Process groups properties

- .NET Core application DLL**
- .NET Core application DLL path
- ASP.NET Core application path
- AWS ECR image account ID
- AWS ECR image region
- AWS ECS cluster
- AWS ECS container name
- AWS ECS family

☒ Apply to all services provided by the process groups

Add a condition
There must be at least one condition

You can add more than one condition
These are treated as an AND statement

Edit a Tag – Inherit and Preview

The screenshot shows the 'Edit a Tag' interface. At the top, there are two checked checkboxes: 'Apply to underlying hosts of matching process groups' and 'Apply to all services provided by the process groups'. Below these is a 'Preview' button. A callout points to the checkboxes, stating: 'The Tag can be inherited to other entities'. Another callout points to the 'Preview' button, stating: 'Select Preview to see what entities would match your tag settings'. Below the 'Preview' button is a section titled 'Matching entities'. It contains a note: 'Preview shows entities that were active and online within the last 72 hours only. Rules will apply to older entities.' Below this note is a tabbed interface with tabs: 'Service(2)', 'Data Cen...', 'Applicati...', 'Synthet...', 'HTTP mo...', 'Infrastruc...', and 'AppMon(0)'. The 'Service(2)' tab is selected and highlighted with a red box. Below the tabs is a table with columns 'Name' and 'Type'. The table contains two entries: 1. 'Dev_DCRUM - /' with type 'Web request service'. 2. 'Dev_DCRUM - Requests executed in background threads of Code*Service.exe' with type 'Background activity'. At the bottom of the 'Matching entities' section are 'Create rule' and 'Cancel' buttons. A callout points to the 'Create rule' button, stating: 'Create the rule once you have used "Preview" to confirm it works as expected'. At the very bottom of the interface is a message: 'You have unsaved changes.' with 'Save changes' and 'Discard changes' buttons. A callout points to the 'Save changes' button, stating: 'Save any changes made to the Tag'.

☒ Apply to underlying hosts of matching process groups

☒ Apply to all services provided by the process groups

Preview

Matching entities

Preview shows entities that were active and online within the last 72 hours only. Rules will apply to older entities.

Service(2) Data Cen... Applicati... Synthet... HTTP mo... Infrastruc... AppMon(0)

Name	Type
Dev_DCRUM - /	Web request service
Dev_DCRUM - Code*Service.exe (No requests for 23h 34min)	
Dev_DCRUM - Requests executed in background threads of Code*Service.exe	Background activity
Dev_DCRUM - Code*Service.exe (No requests for 23h 33min)	

Create rule Cancel

You have unsaved changes.

Save changes Discard changes

The Tag can be inherited to other entities

Select Preview to see what entities would match your tag settings

Create the rule once you have used "Preview" to confirm it works as expected

Save any changes made to the Tag

Edit a Tag – Multiple Rules

Tag name

Application

Add a new rule

Filter by rule name

Rule	Delete	Disable/Enable	Edit
Process groups get value '{HostGroup:Name/^[^_]+_[^_]+})' where Kubernetes container name does not exist and that run on Hosts where Host group name contains regex '^[^_]+_[^_]+}'	X	<input checked="" type="checkbox"/>	✓
Web applications get value '{WebApplication:Name/^[^_]+_[^_]+})' where Web application name contains regex '^[^_]+_[^_]+}' ...	X	<input checked="" type="checkbox"/>	✓
Browser monitors get value '{BrowserMonitor:Name/^[^_]+_[^_]+})' where Browser monitor name contains regex '^[^_]+_[^_]+}'	X	<input checked="" type="checkbox"/>	✓
HTTP monitors get value '{HttpMonitor:Name/^[^_]+_[^_]+})' where HTTP monitor name contains regex '^[^_]+_[^_]+}'	X	<input checked="" type="checkbox"/>	✓
Process groups get value '{ProcessGroup:KubernetesContainerName}' where Kubernetes container name exists	X	<input checked="" type="checkbox"/>	✓

Multiple rules can be added to a tag.
These are treated as an OR statement.

Importing Metadata and Tags

Automatic Tagging recommendations

- There may be existing tags or metadata in 3rd-party systems that you rely on and that you want to ensure are the same across your organization in an automated fashion.
- Dynatrace recommends providing such metadata and tags via the different means at the time of deployment
 - Kubernetes annotations: <https://www.dynatrace.com/support/help/shortlink/kubernetes-tagging#recommendation>
 - Openshift Deployments: <https://www.dynatrace.com/support/help/shortlink/openshift-tagging#recommendation>
 - Cloud Foundry Metadata: <https://www.dynatrace.com/support/help/shortlink/cloud-foundry-tags#recommendation>
 - AWS tags: <https://www.dynatrace.com/support/help/shortlink/aws-tagging#add-tags-in-aws>
 - Azure tags: <https://www.dynatrace.com/support/help/shortlink/waf-policy#monitor-resources-based-on-tags>
 - Google Cloud Metadata: <https://www.dynatrace.com/support/help/shortlink/google-cloud-hub>
- Any tags will be “read only” in Dynatrace and therefore you won't be able to configure them via the Dynatrace UI.

Automatic Tagging and Annotations with Kubernetes

- Dynatrace also detects Kubernetes properties and annotations that can be used when specifying automated rule-based tags.
- Dynatrace detects the following properties which can be used for automated rule-based tags and property-based process group detection rules
 - Kubernetes base pod name
 - Kubernetes container
 - Kubernetes full pod name
 - Kubernetes namespace
 - Kubernetes pod UID

Technologies Process group HipsterShop: adservice

HipsterShop: adservice

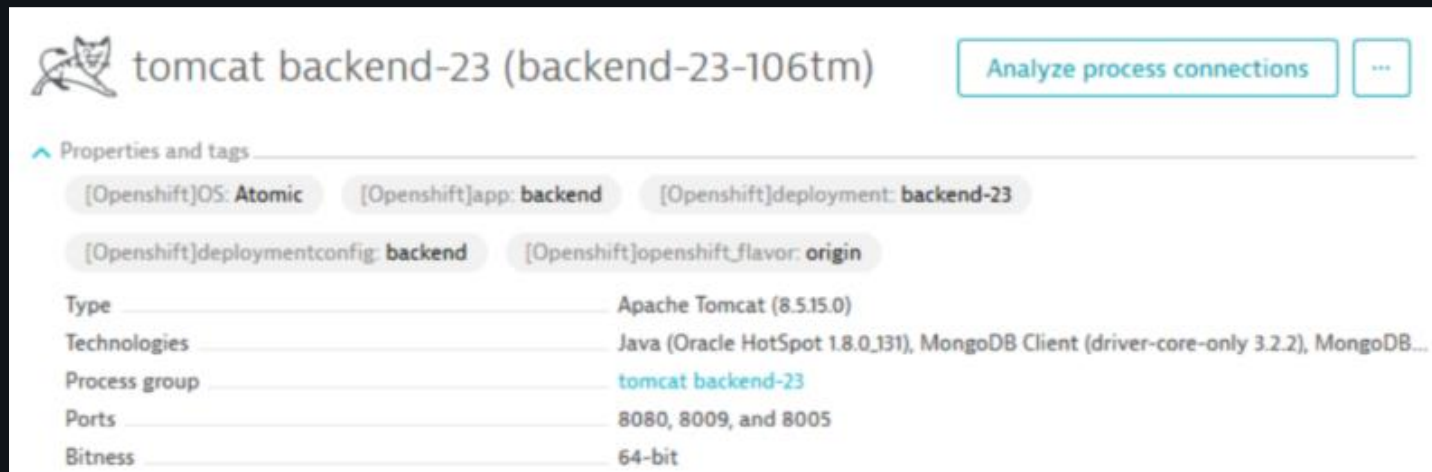
✓ Properties and tags

- [Kubernetes]app: adservice-primary
- [Kubernetes]app.kubernetes.io/instance: adservice-canary
- [Kubernetes]app.kubernetes.io/managed-by: Helm
- [Kubernetes]app.kubernetes.io/name: adservice-primary
- [Kubernetes]app.kubernetes.io/part-of: keptn-hipstershop-prod
- [Kubernetes]app.kubernetes.io/version: jdk11
- [Kubernetes]istio.io/rev: default
- [Kubernetes]security.istio.io/tlsMode: istio
- [Kubernetes]service.istio.io/canonical-name: adservice-primary
- [Kubernetes]service.istio.io/canonical-revision: jdk11
- [Kubernetes]namespace: hipstershop-prod
- k8s-namespace: hipstershop-prod
- keptn_deployment: primary
- keptn_project: hipstershop
- keptn_service: adservice
- keptn_stage: prod
- kubernetes-namespace: hipstershop-prod
- kubernetes-pod: adservice-primary-*

Detected name	hipstershop.AdService adservice-primary-*
Main technology	Java (OpenJDK 11.0.8)
Secondary technologies	Docker, gRPC (1.17.0), and OpenCensus (0.18.0)
Bitness	64-bit
Command line args	/usr/local/openjdk-11/bin/java -Dlog4j2.contextDataInjector=io.opencensus.contrib.logcorrelation...
Docker full image name	gcr.io/dynatrace-demoability/adservice:jdk11
EXE name	java
EXE path	/usr/local/openjdk-*/bin/java, /usr/local/openjdk-11/bin/java
Extension technology	JAVA
Kubernetes base pod name	adservice-primary-*
Kubernetes cluster	Keptn
Kubernetes container name	server
Kubernetes full pod name	adservice-primary-97db5fcf9-48rdw
Kubernetes namespace	hipstershop-prod
Kubernetes namespace labels	istio-injection:enabled
Kubernetes pod UID	4fc41304-9379-4e4
Kubernetes workload	adservice-primary
Management zones	Keptn: hipstershop, Google Hipster Shop, and Google Cloud
Ports	9555
Kubernetes annotations	
prometheus.io/path	/stats/prometheus
prometheus.io/port	15020
prometheus.io/scrape	true
sidecar.istio.io/status	{"version": "e2cb9d4837cda9584fd272bfa1f348525bcaacfadb7e9b9efbd21a3bb44ad7a1","initContain...

Metadata in Openshift

- It's recommended that you define additional metadata at the deployed system.
- For Kubernetes-based applications, you can simply use Kubernetes annotations.
- Dynatrace automatically detects and retrieves all Kubernetes and OpenShift annotations for pods that are monitored with a OneAgent code module.
- This enables you to use automated tagging rules based on existing or custom metadata
- Tags can be imported directly from Openshift processes as well



The screenshot displays the Dynatrace interface for a specific process. At the top, the process is identified as 'tomcat backend-23 (backend-23-106tm)' with a cat icon. To the right are buttons for 'Analyze process connections' and a menu icon. Below this, a section titled 'Properties and tags' shows several OpenShift-related tags: '[Openshift]OS: Atomic', '[Openshift]app: backend', '[Openshift]deployment: backend-23', '[Openshift]deploymentconfig: backend', and '[Openshift]openshift_flavor: origin'. A table below lists process details:

Type	Apache Tomcat (8.5.15.0)
Technologies	Java (Oracle HotSpot 1.8.0_131), MongoDB Client (driver-core-only 3.2.2), MongoDB...
Process group	tomcat backend-23
Ports	8080, 8009, and 8005
Bitness	64-bit

Metadata in Cloud Foundry

- CloudFoundry app properties will be detected and the information imported via a Dynatrace Cloud Foundry service that's bound to your application.
- Additionally, Dynatrace automatically detects tags that are provided in the Cloud Foundry environment variables for applications

The screenshot shows the Dynatrace interface for a Cloud Foundry application. At the top, there's a header with the Dynatrace logo and the application name 'CF tomcat easytravel-backend-demo*'. Below this, a section titled 'Properties and tags' contains two tabs: 'CFAppID: 071dd26f-52c6-482b-bbae-ddd03c05f4a5' and 'cloudfoundry-space: demo-2'. The main content is a table listing various properties and their values.

Property	Value
Detected name	tomcat easytravel-backend-demo*
Main technology	Apache Tomcat (9.0.16.0)
Secondary technologies	Apache HTTP client (4.5.10), Garden, Java (OpenJDK 1.8.0_202), MongoDB Client
Bitness	64-bit
Catalina base	/home/vcap/app/java-buildpack/tomcat
Catalina home	/home/vcap/app/java-buildpack/tomcat
Cloud Foundry application	easytravel-backend-demo*
Cloud Foundry application ID	071dd26f-52c6-
Cloud Foundry instance index	0
Cloud Foundry organization	demo
Cloud Foundry space	demo-2
Cloud Foundry space ID	362a9aed-7f
Command line args	/home/vcap/app/java-buildpack/open_jdk_jre/bin/java -Djava.util.logging.config
Container ID	4f75b5cb-80d2-40
EXE name	java
EXE path	/home/vcap/app/java-buildpack/open_jdk_jre/bin/java
Extension technology	TOMCAT
Host group	demo-2
Management zones	Cloud Foundry and CustomerC
Ports	8080

Automatic Tagging with AWS

- AWS Tags will be automatically imported
- By default, Dynatrace monitors all Amazon Web Services that have been specified in your permission policy.
- Tags can allow leveraging multiple Dynatrace environments to monitor distinct AWS services running under the same AWS account.
- Optionally, you can use tagging to limit the AWS resources (AWS service instances) that are monitored by Dynatrace.

The screenshot displays the Dynatrace interface for a monitored instance. At the top, the instance is identified as 'BB1-apache-tomcatjms-iis' with an uptime of 'over 574 days'. A button labeled 'Analyze process connections' is visible in the top right corner. Below the instance name, the 'Properties and tags' section is expanded, showing various attributes and their values. These include tags like '[AWS]APL.Deployment', '[AWS]APL.Environment', '[AWS]APL.Status', '[AWS]DetailUsage', '[AWS]Email', '[AWS]MonitoredBy', '[AWS]Name: et-demo-2-win1', '[AWS]ShouldBeReserved: true', '[AWS]Usage: APL', '[AWS]monitor_demo1: false', and '[AWS]monitor_demo2: true'. There are also buttons for 'ServiceNow' and 'Teste'. Below the tags, a table lists system and cloud properties for the instance.

Property	Value
Detected name	demo-2-win-1
OneAgent version	1.211.144.20210305-115326
Technologies	Citrix and Citrix Xen
AMI ID	ami-fc0fc94
AWS name tag	et-demo-2-win1
Architecture	x86, 64-bit
Availability Zone	us-east-1b
Cloud	EC2
Instance ID	i-0123456789abcdef0
Instance type	m3.large
Logical CPU cores	2
Management zones	Windows hosts, Citrix Overview, and Production
Monitoring mode	Full stack
Physical CPU cores	1
Private IP address	10.0.0.1
Private host name	demo-2-win-1
Public IP address	54.191.100.1
Public host name	demo-2-win-1
Security group	sg-12345678
Virtualization	Xen

Automatic Tagging with Azure

- Configure Azure Tag Capture when defining the Azure Connection

Azure Azure ServicePrincipal for DemoEnvironment-Dev

Connection name

Client ID

Tenant ID

Secret Key

Resources to be monitored

☒ Capture Azure tags automatically

Supporting services
No supporting services are monitored

LoadGen Uptime: over 977 days

✓ Properties and tags

[Azure]ACE:CREATED-BY: [Azure]Category: DEV

[Azure]Owner: [Azure]clusterName: [Azure]resourceType: Service Fabric

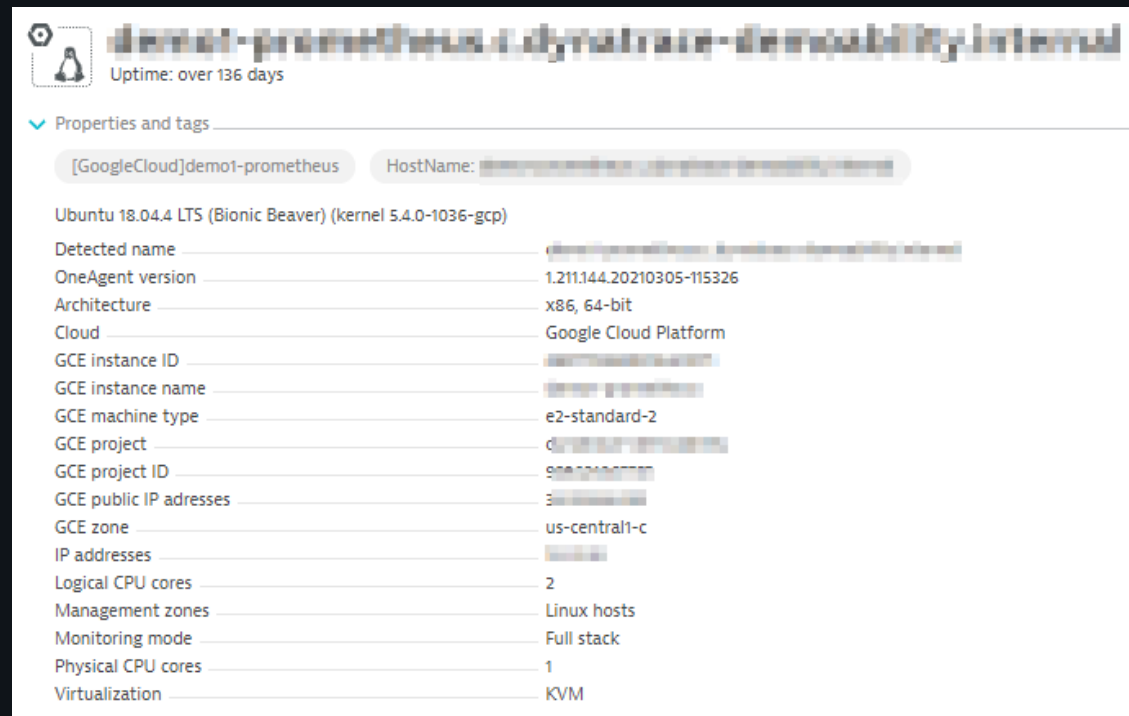
CLOUD_TYPE_EQ_AZURE Cloud_Azure: southcentralus HostName: [Azure]tenant: CustomerA

Windows Server 2016 Datacenter 1607, ver. 10.0.14393

Detected name	
OneAgent version	1.211.144.20210305-115326
Architecture	x86, 64-bit
Azure scale set	
Cloud	Azure
IP addresses	
Instance type	Standard_D1_v2
Logical CPU cores	1
Management zones	Windows hosts, Azure,
Monitoring mode	Full stack
Physical CPU cores	1
Region	southcentralus
Virtualization	Hyper-V

Metadata and Tags with Google Cloud Platform (GCP)

- Metadata and Tags will be imported for GCP monitored hosts



demo1-prometheus Uptime: over 136 days

▼ Properties and tags

[GoogleCloud]demo1-prometheus HostName: demo1-prometheus

Ubuntu 18.04.4 LTS (Bionic Beaver) (kernel 5.4.0-1036-gcp)

Detected name	demo1-prometheus
OneAgent version	1.211.144.20210305-115326
Architecture	x86, 64-bit
Cloud	Google Cloud Platform
GCE instance ID	demo1-prometheus
GCE instance name	demo1-prometheus
GCE machine type	e2-standard-2
GCE project	demo1-prometheus
GCE project ID	demo1-prometheus
GCE public IP addresses	demo1-prometheus
GCE zone	us-central1-c
IP addresses	demo1-prometheus
Logical CPU cores	2
Management zones	Linux hosts
Monitoring mode	Full stack
Physical CPU cores	1
Virtualization	KVM

Naming Rules

Customize Hosts, Process Groups and Services via rules

- Dynatrace automatically detects and names entities when monitored.
- It's common in environments for detected names to be unintuitive or even confusing. Different teams may refer to hosts by different properties or components they support.
- Dynatrace supports renaming via automated rules.
- Naming rules allow you to use a wide set of metadata, or properties, to adapt the default names of automatically discovered hosts, process groups or services.
 - Any property that is displayed within the Properties & Tags section can be used
 - Portions of host, process groups or service names can be included using regular expressions
- When a rule is in place any existing entity names are changed using the naming rule.

Host Naming Rules

Creating Host Naming Rules

- Settings -> Monitoring -> Host Naming
- Multiple rules are processed from the top. The first rule with matching conditions is used. The remaining rules are skipped.
- Rules can be reordered, enabled/disabled and edited.
- -> Click add rule.

The screenshot shows the 'Host naming' configuration page. On the left is a sidebar with a 'Settings' menu. The 'Monitoring' section is expanded, showing 'Setup and overview', 'Monitored technologies', 'Monitoring overview', 'Host naming' (selected), 'Windows services availability', and 'SLOs'. Below this, the 'Processes and containers' section is also expanded, showing 'Detection and naming', 'Processes', 'Process group monitoring', 'Process group detection', 'Process group naming', and 'Declarative Process Grouping'. The main content area is titled 'Host naming' and includes a description: 'Naming rules enable you to customize and enhance the default names of your hosts. [More...](#)'. Below this is a section for 'Custom naming rules' with an 'Add a new rule' button. A search bar labeled 'Filter by rule name' is present. A table lists the existing rules with columns for 'Rule', 'Move up/down', 'Delete', 'Disable/Enable', and 'Edit'.

Rule	Move up/down	Delete	Disable/Enable	Edit
Prefix with HG	▼	×	<input type="radio"/>	▼
Host TAG rename rule with Host Group	▲ ▼	×	<input checked="" type="checkbox"/>	▼
Host TAG rename rule	▲ ▼	×	<input checked="" type="checkbox"/>	▼
Host and OS with Host Group	▲ ▼	×	<input checked="" type="checkbox"/>	▼
Host and OS	▲	×	<input checked="" type="checkbox"/>	▼

Creating Host Naming Rules

- Provide a Rule Name
- Define the name format. This can include any static text and optional placeholders.
 - Regex can be used against selected properties {Env-Var/regex}
 - Try to create efficient regular expressions as per:
<https://www.dynatrace.com/support/help/shortlink/regex>
- If necessary, limit the scope to entities with certain properties
- Add one or more conditions to identify the hosts targeted for renaming and to verify a property exists for use in the name.

Host naming

Naming rules enable you to customize and enhance the default names of your hosts. [More...](#)

Custom naming rules

Add a new rule

Hosts where Host group name exists

Rule name

Add hostgroup to the name

Host name format

{HostGroup:Name} - {Host:DetectedName}

⚠ Specified placeholders not applicable to matching Hosts will resolve to an empty string.

Rule applies to entities matching the following properties

All host groups

All technologies

Conditions

Host group name

exists

Remove condition

Creating Host Naming Rules

- Preview results to ensure the matching entities are correct and the name format is what is desired
- Create the rule and save the changes











- For more information:

<https://www.dynatrace.com/support/help/shortlink/hostname-config>

Preview

Matching entities

Preview shows entities that were active and online within the last 72 hours only. Rules may still apply to older entities.

Name	New name
 weblauncher-et-demo2 	vmware - weblauncher-et-demo2
 weblauncher-demo.c.dynatrace-demoability.internal 	google_cloud - weblauncher-demo.c.dynatrace-demoability.internal
 SAP-A-PAS.lab2.net 	SAP - SAP-A-PAS.lab2.net
 SAP-A-DB.lab2.net 	SAP - SAP-A-DB.lab2.net
 SAP-A-ASCS.lab2.net 	SAP - SAP-A-ASCS.lab2.net

< 1 2 3 4 5 6 >

Create rule

Cancel

Process Group Naming

Process Group Naming

- Naming rules apply to already detected process groups
- If you need to change the makeup of process groups, please use Process group detection rules instead

Creating Process Group Naming Rules

- Settings -> Processes and containers -> Process group Naming
- Multiple rules are processed from the top. The first rule with matching conditions is used. The remaining rules are skipped.
- Rules can be reordered, enabled/disabled and edited.
- -> Click add rule.

Settings > **Processes and containers** > **Process group naming**

Settings

- Monitoring
 - Setup and overview
 - Monitored technologies
 - Monitoring overview
 - Host naming
 - Windows services availability
 - SLOs
- Processes and containers**
 - Detection and naming
 - Processes
 - Process group monitoring
 - Process group detection
 - Process group naming**
 - Declarative Process Grouping

Process group naming

Naming rules enable you to customize and enhance the default names of your deployed process groups. Naming rules apply to already detected process groups and processes that are active and online. [More...](#)

Custom naming rules

[Add a new rule](#)

Filter by rule name

Rule	Move up/down	Delete	Disable/Enable	Edit
DT Windows JRE Cleanup	▼	×	<input type="checkbox"/>	▼
K8s add Namespace	▲ ▼	×	<input checked="" type="checkbox"/>	▼
Host Group prefix	▲	×	<input checked="" type="checkbox"/>	▼

Creating Process Group Naming Rules

- Provide a Rule Name
- Define the name format. This can include any static text and optional placeholders.
 - Regex can be used against selected properties {Env-Var/regex}
 - Try to create efficient regular expressions as per:
<https://www.dynatrace.com/support/help/shortlink/regex>
- If necessary, limit the scope to entities with certain properties
- Add one or more conditions to identify the process groups targeted for renaming and to verify a property exists for use in the name.

Process group naming

Naming rules enable you to customize and enhance the default names of your deployed process groups. Naming rules apply to already detected process groups and processes that are active and online. [More...](#)

Custom naming rules

Add a new rule

Process groups on **Hosts** where Host group name exists

Rule name

Add hostgoup to the name

Process group name format

{HostGroup:Name} - {ProcessGroup:DetectedName}

⚠ Specified placeholders not applicable to matching Process groups will resolve to an empty string.

Rule applies to entities matching the following properties

All host groups

All technologies

Conditions

Host group name

exists

Remove condition

Add condition

Creating Process Group Naming Rules

- Preview results to ensure the matching entities are correct and the name format is what is desired
- Create the rule and save the changes











- For more information:



<https://www.dynatrace.com/support/help/shortlink/process-group-naming>

Preview

Matching entities

Preview shows entities that were active and online within the last 72 hours only. Rules may still apply to older entities.

Name	New name
 SAP host agent 	SAP - SAP host agent
 python3 downloads-* download-server 	demo-2 - python3 downloads-* download-server
 prom-label-proxy thanos-querier-* 	demo-2 - prom-label-proxy thanos-querier-*
 prom-label-proxy prometheus-k*s-* 	demo-2 - prom-label-proxy prometheus-k*s-*
 opm community-operators 	demo-2 - opm community-operators

 1 2 3 ... 9 10 

Service Naming Rules

Creating Service Naming Rules

- Settings -> Server-side service monitoring -> Service naming rules
- Multiple rules are processed from the top. The first rule with matching conditions is used. The remaining rules are skipped.
- Custom rules can be reordered, enabled/disabled and edited.
- Custom service naming rules override built-in rules.
- Built-in rules are listed but cannot be changed.
- -> Click add rule.

The screenshot shows the 'Service naming rules' configuration page. The left sidebar contains a 'Settings' menu with categories like Monitoring, Processes and containers, Web and mobile monitoring, Cloud and virtualization, and Server-side service monitoring. The 'Server-side service monitoring' section is expanded, showing 'Service naming rules' as the selected option. The main content area is titled 'Service naming' and includes a description: 'Naming rules enable you to customize and enhance the default names of your services.' Below this, there is a section for 'Custom naming rules' with an 'Add a new rule' button. A table lists the custom rules: 'K8s Add Namespace' and 'Host Group Prefix'. Each rule has controls for reordering (Move up/down), deleting (Delete), enabling/disabling (Disable/Enable), and editing (Edit). The 'Host Group Prefix' rule is currently enabled. Below the custom rules is a section for 'Built-in rules', which lists several predefined rules like 'PHP web request service (user defined application ID)' and 'Web request service (with web application ID)'. These rules also have enable/disable and edit controls.

Settings > **Server-side service monitoring** > **Service naming rules**

Settings

- Monitoring
 - Setup and overview
 - Monitored technologies
 - Monitoring overview
 - Host naming
 - Windows services availability
 - SLOs
- Processes and containers
 - Detection and naming
- Web and mobile monitoring
 - Real user and synthetic monitoring
- Cloud and virtualization
 - Connect cloud and virtualization types
- Server-side service monitoring**
 - Manage and customize service monitoring
 - Custom service detection
 - Merged service monitoring
 - Service naming rules**
 - Request attributes
 - Calculated service metrics
 - Request naming

Service naming

Naming rules enable you to customize and enhance the default names of your services.
[More...](#)

Custom naming rules

[Add a new rule](#)

Filter by rule name

Rule	Move up/down	Delete	Disable/Enable	Edit
K8s Add Namespace	▼	✕	<input checked="" type="checkbox"/>	✓
Host Group Prefix	▲	✕	<input checked="" type="checkbox"/>	✓

Built-in rules

Rule	Disable/Enable	Details
PHP web request service (user defined application ID)	<input type="checkbox"/>	✓
PHP web request service	<input type="checkbox"/>	✓
Web request service (with web application ID)	<input type="checkbox"/>	✓
Web request service (without web application ID)	<input type="checkbox"/>	✓
Web request service (with web application ID on a process that isn't deeply monitored)	<input type="checkbox"/>	✓
Web request service (on a process that isn't deeply monitored)	<input type="checkbox"/>	✓

Creating Service Naming Rules

- Provide a Rule Name
- Define the name format. This can include any static text and optional placeholders.
 - Regex can be used against selected properties `{Env-Var/regex}`
 - Try to create efficient regular expressions as per:
<https://www.dynatrace.com/support/help/shortlink/regex>
- If necessary, limit the scope to entities with certain properties
- Add one or more conditions to identify the services targeted for renaming and to verify a property exists for use in the name.

Service naming

Naming rules enable you to customize and enhance the default names of your services.

[More...](#)

Custom naming rules

Add a new rule

Services on Hosts where Host group name exists

Rule name

Add hostgroup to the name

Service name format

{HostGroup:Name} - {Service:DetectedName}

⚠ Specified placeholders not applicable to matching Services will resolve to an empty string.

Rule applies to entities matching the following properties

All process groups



All host groups



All service types



All technologies



All service topologies



Conditions

Host group name



exists



Remove condition

Add condition

Creating Service Naming Rules

- Preview results to ensure the matching entities are correct and the name format is what is desired
- Create the rule and save the changes






- For more information:

<https://www.dynatrace.com/support/help/shortlink/custom-service-names>

Preview

Matching entities

Preview shows entities that were active and online within the last 72 hours only. Rules may still apply to older entities.

Name	Type	New name
 BookingService eT-vmware-demo-2-BusinessBackend	Web service	vmware - BookingService
 ConfigurationService openshift-easytravel-backend	Web service	demo-2 - ConfigurationService
 EasytravelService com.dynatrace.easytravel.customer.frontend.rest.jar easytravel-*-x*	Web service	google_cloud - EasytravelService
 JourneyService openshift-easytravel-backend	Web service	demo-2 - JourneyService
 RESTProcedureControl com.dynatrace.easytravel.cmdlauncher.jar easytravel-*-x*	Web service	openstack - RESTProcedureControl

< 1 2 3 ... 9 10 >

Management Zones

Why Management Zones?

- Easily focus on relevant entities
 - Example: Multiple application environments being monitored within the same Dynatrace Environment
 - A management zone filter can be applied to only view Production or Dev Applications, Services, Processes, or hosts
- A management zone consists of multiple rules that define the entities that are included
 - Rules are built upon matching entities and the powerful Dynatrace tagging engine
- Management zones are available as **filters** that are accessible via the menu bar
 - Management zone filters apply to all views that display multiple entities, such as Dashboards, list pages (hosts, process groups, services, and applications), Smartscape, and the Technologies page
- Limit group access and **permissions** to part of the monitoring data

Creating a Management Zone

- Settings -> Preferences -> Management Zones

Preferences

Environment settings

OneAgent updates

Data privacy and security

Management zones

Create management zone

To define a management zone, you must set up at least one rule that specifies what is to be included in the management zone.

Management zone name

For example, Production or Dev

Add new rule

Apply Rules Matching the desired 'Zone'

- Rules can be applied in the scope of Services, Process Groups, Hosts, Custom Devices, Applications (Web, Enterprise, Mobile), Synthetic monitors, AWS, Azure, Cloud Foundry, and Metrics.
- Choose a scope above and apply one or more rules to match the appropriate entities.
- Many different combinations of rules can be used, Tags can also be selected as a drop down.
- A single rule with multiple conditions acts as logical AND of those conditions/properties.
- One or more rules will act as a logical OR between rules for that management zone.

Set up Management Zones

Settings

Preferences

Management zones

Staging

Settings

Monitoring

Monitored technologies

Monitoring overview

Host naming

Processes and containers

Web and mobile monitoring

Cloud and virtualization

Server-side service monitoring

Custom service detection

Merged service monitoring

Service naming rules

Edit management zone

Management zone name

Staging

Add a new rule

Filter by rule name

Rule	Delete	Disable/Enable	Edit
Hosts where VMware data center name begins with 'eTravel-demo'	X	<input checked="" type="checkbox"/>	▼
Services on hosts where VMware data center name begins with 'eTravel-demo'	X	<input checked="" type="checkbox"/>	▼
Web applications where Web application name equals 'www.vmware.easytravel.com'	X	<input checked="" type="checkbox"/>	▼
Browser monitors where Browser monitor name begins with 'www.vmware'	X	<input checked="" type="checkbox"/>	▼

Matching entities

Service(42)

Custom Dev...

Custom Dev...

Data Center...

Application ...

Synthetic m...

HTTP monit...

Infrastructu...

AppMon(0)

Azure(0)

Name	Type
AuthenticationService eT-vmware-demo-l-BusinessBackend	Web service
azure RESTProcedureControl com.dynatrace.easytravel.cmdlauncher.jar easytravel-*-x*	Web service
BookingService eT-vmware-demo-l-BusinessBackend	Web service

Select a management zone to focus on

- Configured Management zones are available to filter on throughout the UI

The screenshot displays the Dynatrace dashboard interface. The top navigation bar includes a search bar with the text "Search Dynatrace demo1...", a filter icon, and a "Last 2 hours" time range selector. The main content area is divided into several sections:

- Home**: The main dashboard area.
- Quick overview**: A section with four cards showing health metrics:
 - Problems**: 0
 - Application health**: All fine, 25
 - Synthetic monitor...**: All fine, 12
 - Database health**: All fine, 21
- Smartscope**: A card showing 25 Applications.
- Application health**: A section with a "Worldmap (Apdex)" showing a map of the United States and a "Service health" section showing a hexagonal grid of green dots representing 196 services, all marked as "All fine".
- Infrastructure**: A section with several cards:
 - Host health**: All fine, 64
 - Network status**: Shows 60 Hosts and a volume of 535 Mbit/s.
 - AWS**: Shows 3 RDS instances, 19 Load balancers, and 493 EC2 instances.
 - Docker**: Shows 9 Docker hosts, 131 Containers, and 48 Images.

On the right side, a dropdown menu titled "Dynatrace demo1" is open, showing a list of management zones. The zones listed are: All, AIX, Azure, Azure EasyTravel, Citrix Overview, Cloud Foundry, CustomerA, CustomerB, CustomerC, ESXi_visibility, easytravel Apps, Google Cloud, Google Hipster Shop, Kubernetes, Linux hosts, mainframe, OpenShift, Production, SAP, SAP&Citrix, **Staging** (highlighted), Test, and Windows hosts. The "Staging" zone is currently selected.

Assign access rights to management zones

- Applicable to users and user group permissions with access rights to specific zones
- In this way, users will only see those entities that are relevant to their work

Management zone permissions		
Filter by		
Environment	Inherit parent permissions	Edit
edu24117::Operations Team	<input checked="" type="checkbox"/>	▼
edu24117::Development Process Groups and Hosts	<input checked="" type="checkbox"/>	▼
edu24117::Business View Allowed	<input checked="" type="checkbox"/>	▼
edu24117::Dev Team	<input checked="" type="checkbox"/>	▼
edu24117::Architect Team	<input checked="" type="checkbox"/>	▼
edu24117::DevHosts	<input checked="" type="checkbox"/>	▼
edu24117::Frontend Services	<input checked="" type="checkbox"/>	▼
faa73910::Kubernetes	<input checked="" type="checkbox"/>	▼
fin07364::Dynatrace University	<input type="checkbox"/>	▼
fsg94658::Keptn: simpleproject dev	<input type="checkbox"/>	▼
Show more		

Questions?



Simply smarter clouds