



# OpenShift Container Platform 4.3

## Managing cost data using tagging

Organize resources and allocate costs with tags



## OpenShift Container Platform 4.3 Managing cost data using tagging

---

Organize resources and allocate costs with tags

## Legal Notice

Copyright © 2020 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

This guide explains how tagging works in cost management, and outlines strategies for managing your cost data with tags and labels.

---

## Table of Contents

<b>CHAPTER 1. MANAGING COST DATA USING TAGGING .....</b>	<b>3</b>
1.1. HOW COST MANAGEMENT ASSOCIATES TAGS	3
1.1.1. Tag matching hierarchy	4
1.1.1.1. Direct resource matching (EC2 instance ID)	4
1.1.1.2. Special OpenShift tags	4
1.1.1.3. Custom tags	5
1.2. DESIGNING YOUR TAGGING STRATEGY	5
1.2.1. Map business to reporting	5
1.2.2. Standardize your labels	6
1.2.3. Tag all elements on your sources (manually or through automation)	6
1.2.4. Review your tags often and refine as needed	6
1.2.5. Select your tag terminology	6
1.3. TAG SPECIFICATIONS BY SOURCE TYPE	7
1.4. ADDITIONAL RESOURCES	8



# CHAPTER 1. MANAGING COST DATA USING TAGGING

This guide explains in depth how tagging works in cost management, and how you can use tagging to best organize your resources to manage your costs.



## NOTE

Cost management is currently Technology Preview. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process.

For more information about the support scope of Red Hat Technology Preview features, see [Technology Preview Features Support Scope](#).

If you have a suggestion for improving this guide or have found an error, please submit a Bugzilla report at <http://bugzilla.redhat.com> against *Cloud Software Services (cloud.redhat.com)* for the **Cost Management** component.

## 1.1. HOW COST MANAGEMENT ASSOCIATES TAGS

Tagging is done in each source before cost management can use the tags to automatically organize your cost data.

Add your tags or labels to resources belonging to each source. Then, cost management imports the tags by different methods, depending on source type:

- AWS tags must be activated, and are then selected and exported to cost management in the Cost and Usage report. See the *Getting started with cost management* guide for instructions on activating AWS tags.
- In OpenShift Container Platform, labels are exported by Metering and included in the metrics reports that cost management uses as input.



## NOTE

See *Getting started with cost management* for instructions on configuring sources and tags.

AWS tags and OpenShift labels both consist of key:value pairs. When the key:value pairs match, the AWS and OpenShift costs are automatically associated by cost management. Note that tag matching is not case sensitive: for example, an AWS resource AWS tagged *APP* and an OpenShift resource tagged *app* are a match:

**Table 1.1. Example: Tag matching**

Source and resource type	Key	Value
AWS Resource (RDS)	APP	Cost-Management
OpenShift pod	app	cost-management

If an AWS resource tag matches with multiple OpenShift projects, the cost and usage of that resource are split evenly between the matched projects.

Note that this is not the case with AWS compute resources that are matched via the instance ID-node relationship. In that case, cost and usage are broken down using information about a project's resource consumption within the OpenShift cluster.

### 1.1.1. Tag matching hierarchy

To identify your OpenShift resources running on AWS or Azure instances, cost management matches tags between sources in the following order:

1. Direct resource matching (EC2 instance ID)
2. Special OpenShift tags
3. Custom tags

#### 1.1.1.1. Direct resource matching (EC2 instance ID)

The sources apply these tags automatically. This form of tagging provides a direct link between AWS EC2 instances and OpenShift nodes.

AWS assigns every EC2 instance a resource identifier (a number such as **i-01f44b3d90ef90055**). OpenShift nodes are matched directly to the AWS EC2 instance the cluster is running on using the AWS resource identifier. The OpenShift reports in cost management (generated from Prometheus data) include this identifier for nodes.

#### 1.1.1.2. Special OpenShift tags

There are three special-case AWS tags you can use to associate cost with OpenShift:

- **openshift\_cluster**
- **openshift\_node**
- **openshift\_project**

These tags have matching priority over custom tags, and are especially useful in differentiating the costs of different OpenShift clusters running on the same AWS instance.

To use this tagging method to identify an OpenShift cluster:

1. Name your OpenShift cluster (in this example, **dev-cluster**).
2. Tag your AWS instance with the key **openshift\_cluster**, and provide the cluster name as the value (in this example, **dev-cluster**).

Table 1.2. Example: Special OpenShift tags

Source and resource type	Key	Value
AWS Resource (RDS)	<b>openshift_cluster</b>	<b>dev-cluster</b>



Source and resource type	Key	Value
OpenShift cluster	No tags needed. Name your OpenShift cluster <b>dev-cluster</b> .	No tags needed. Name your OpenShift cluster <b>dev-cluster</b> .

### 1.1.1.3. Custom tags

You can use any key:value combination as tags, and cost management will associate identical tag key and values together. You can then group costs by tag key, account, service, region, and more to view your costs and charge for that tag.

**Table 1.3. Example: Custom tags**

Source and resource type	Key	Value
AWS Resource (RDS)	<b>team</b>	<b>engineering</b>
OpenShift pod	<b>team</b>	<b>engineering</b>

## 1.2. DESIGNING YOUR TAGGING STRATEGY

When designing your tagging strategy, there are several considerations for determining how to organize and report costs for your sources.

### 1.2.1. Map business to reporting

Define the perspectives you want to report on. For example, your taxonomy for cost management could consider these different perspectives:

#### Ownership and usage

The owner and the user of the resource: for example, the unique identifier of the user who requested the resource and the one that is actually consuming the resource.

#### Tenancy

If your environment is shared, it can be beneficial to understand which group or business unit has requested the resource. When the user is part of different groups, one needs to be selected. For cost reporting, this is achieved in many cases using cost center, but department, project, partner are also good candidates.

#### Location

For organizations with software deployed globally, cloud providers already identify the region where your resources are running, but your private cloud may be different.

#### Environment or stage

You may want to differentiate between development and production, so that different cost decisions can be made depending on the environment where you are creating or running the resources. If your development pipeline already includes stages, like development, testing, staging, pre-production and/or production, this is a good candidate.

#### Application / Project / Service / Event

What is the service that your resource is helping to provide? Is that a group of transient resources for an event (your yearly customer-focused demo show?). You could even include the application version.

### 1.2.2. Standardize your labels

Consistency is the most important aspect of a tagging strategy to deliver accurate and comparable results.

Create a clear tagging policy defining what needs to be tagged, what tags are mandatory and what tags are optional, making sure that there is no room for interpretation.

If the values need to be chosen between a list, make sure that those values are defined, consistent, and easily accessible, or that the list is presented to the user. For example, if you are defining development with the key "Development", do not also use "Dev", "DEV", or "R&D" to also identify resources as "Development".

### 1.2.3. Tag all elements on your sources (manually or through automation)

Since untagged resources cannot be reported, tag as many elements as possible, ideally using automation to prevent human error. Sources have different automation features to take advantage of for tagging:

- In Azure, you can use Azure Policies to enforce tagging rules and conventions and avoid resources being deployed that do not comply with your expectations. You can create a policy that automatically applies the needed tags during provisioning, that enforce a predefined format for dates, or that just make some tags mandatory for some resource type.
- In AWS, you can use IAM policies for the same. And you can use an automation tool like Ansible to add the necessary tags during provisioning and make sure that all the resources have been properly tagged.
- OpenShift Container Platform does not presently have a method of automating labelling.

### 1.2.4. Review your tags often and refine as needed

Define your tags and use them as early as possible with cost management, even if you need to adjust your tagging scheme afterwards.

Review the resulting reports with your business owner/stakeholders early on to ensure your tags are helping you to generate the desired reports, and review your tagging strategy every few weeks to optimize.

### 1.2.5. Select your tag terminology

- Name your resources with names that allow you to identify your resources without accessing metadata (many clouds have a guide about how to do this properly; see Additional resources for links), and then continue by adding metadata to it.
- Map your resources into keys and values. Keys will map to perspectives, while values will define the different options allowed for each key. In some cases, the value will be Null.

**NOTE**

Not all sources allow the same identifiers, and have different limitations. See the *Tag specifications by source type* table in this guide for limitations by source.

### 1.3. TAG SPECIFICATIONS BY SOURCE TYPE

Tagging standards differ between source types. To use the same tags/labels across sources, you must use the most common of all the restrictions across the different providers.

The following table summarizes tagging and labelling criteria across AWS, Azure and OpenShift Container Platform 4:

**Table 1.4. Tagging specifications by source**

Criteria	AWS	Azure	Red Hat OpenShift
Name	Tags	Tags	Labels
Format	Key & value	Name & value	Key & value Keys: [prefix/]name Prefix: must be a DNS subdomain
Allows empty value	Yes	Yes	Yes
Unique label per key	Yes	Yes	Yes
Case sensitive	Yes	No	Yes
Limit per resource	50	50 / 15 for storage	N/A
Length of key	128	512 (128 for storage)	253(prefix) / 63(name)
Length of value	256	256	63
Allowed characters	Letters, numbers, and spaces representable in UTF-8, and the following characters: + - = . _ : / @	Tag names cannot contain these characters: <, >, %, &, \, ?, /	The name segment is required and must be 63 characters or less, beginning and ending with an alphanumeric character ([a-z0-9A-Z]) with dashes (-), underscores (_), dots (.), and alphanumerics between

Criteria	AWS	Azure	Red Hat OpenShift
Restrictions	The prefix "aws:" is reserved. Tags applied to EC2 can use any character. Not all resource types support tags.	Not all resource types support tags. Generalized VM don't support tags. Tags applied to the resource group are not inherited by the resources. Tags can't be applied to classic resources.	Prefixes kubernetes.io/ And k8s.io/ are reserved. Not all resource types support tags.
Notes	You need to select the tags keys that will be included in cost and usage files and billing reports.	You can use a JSON string to go over the limit of keys.	If the prefix is omitted, the label Key is presumed to be private to the user.

## 1.4. ADDITIONAL RESOURCES

The following links provide further guidance on tagging for each source type.

### AWS:

- [AWS tagging strategies](#)
- [IAM: Add a specific tag with specific values](#)

### OpenShift:

- [Kubernetes labels and selectors](#)
- [Kubernetes user guide: labels](#)

### Microsoft Azure:

- [Azure resource naming and tagging decision guide](#)
- [Azure naming best practices](#)
- [Using tags to organize your Azure resources](#)
- [Enforce tags in Azure resource groups](#)