

TIBCO® Messaging Manager

Apache Pulsar™ Command Reference

Version 4.0.0 | March 2025

Contents

Contents	2
About this Product	5
Getting Started	6
MSGMX Apache Pulsar Management and Commands	8
4lw (Apache ZooKeeper-specific)	11
cd	13
clear-backlog	14
clear-backlog namespace	14
connect	15
create	17
create namespace	17
create schemas	18
create tenant	18
create topic	18
create subscription	19
create cluster	21
delete	22
delete bookie-rack	22
delete cluster	22
delete namespace	22
delete schemas	23

delete subscription	23
delete tenant	23
delete topic	24
disconnect	25
help	26
list	27
list bookies	27
list brokers	27
list clusters	27
list connections	28
list failure-domains	28
list namespace-isolation-policies	28
list namespaces	28
list permissions	28
list resource-groups	29
list subscriptions	29
list tenants	29
list topics	29
ls	30
pwd	31
reset	32
reset backlog-quota-check	32
reset broker	32
reset cluster	33
reset namespace	34
reset subscription	36
reset topic	36

set	38
set bookie	38
set broker	38
set cluster	39
set namespace	42
set subscription	46
set tenant	46
set topic	47
show	48
show bookie-rack	48
show broker	48
show cluster	49
show namespace	50
show racks-placement	52
show schemas	52
show tenant	52
show topic	53
unload	54
unload namespace	54
unload topic	54
TIBCO Documentation and Support Services	56
Updated Resources on TIBCO Community	56
Legal and Third-Party Notices	58

About this Product

TIBCO® is proud to announce the latest release of this TIBCO® Messaging Manager software component.

This release is the latest in a long history of TIBCO products that leverage the power of Information Bus® technology to enable truly event-driven IT environments. To find out more about how TIBCO Messaging Manager software and other TIBCO products are powered by TIB® technology, please visit us at www.tibco.com.

TIBCO Messaging Manager software is part of TIBCO Messaging.

Product Overview

TIBCO Messaging Manager (MSGMX) provides an intelligent, predictive, and auto-completing command-line interface for messaging/streaming systems such as TIBCO Enterprise Message Service (EMS), Apache Kafka, and Apache Pulsar. MSGMX runs in any standard console window or terminal emulator.

Getting Started

Before using MSGMX to manage Apache Pulsar, review the following information and refer to the MSGMX User Guide.

Starting MSGMX to Manage Apache Pulsar

Perform the following steps:

1. Confirm you have at least one Apache Pulsar instance running.
2. In a local console window, either add the directory containing the MSGMX executable file to your path, or navigate to the directory containing the executable. For example:

Default Unix location :

```
cd /opt/tibco/msgmx/bin
```

3. Launch the MSGMX program using the `msgmx` command.
4. Use the `manage` command to select the component to manage. For example, to manage Apache Pulsar:

```
manage pulsar
```

5. At the MSGMX prompt, initiate a connection to an Apache Pulsar instance using the `connect` command:

```
connect Instance1 localhost:8080
```



Note: 8080 is the default port for an Apache Pulsar instance; be sure to specify a host and port that is appropriate for your running Apache Pulsar system.

When the connection is established, the MSGMX prompt changes to `/>`. Verify that the prompt changes, indicating that a connection has been established.

If MSGMX cannot connect using the information you provided, it displays an error

message. Ensure that the target Apache Broker is running and is accessible at the specified host and port.

6. As a test command, enter:

```
list connections
```

You see a listing of your current Apache Pulsar Instance connections.

MSGMX Apache Pulsar Management and Commands

TIBCO® Messaging Manager provides a comprehensive and efficient command interface for administering an Apache Pulsar deployment.

Introduction

The Apache Pulsar commands supported by MSGMX have an intuitive correlation with native Apache Pulsar command combinations and tasks while providing a flexible and assistive interface that frees you from the arcane syntax and limitations of using the native Apache Pulsar scripts. This interface lets you efficiently and effectively manage a complex deployment regardless of the number of tenants, namespaces, or other configuration objects it supports.

Bottom Status Bar

The bottom status bar displays one of several status messages providing key information about the currently managed deployment.

- Number of connected Apache Pulsar instances
- Number of Apache Pulsar clusters under the current instance
- Number of Global ZooKeepers (configuration-store) for the current instance
- Number of Local ZooKeepers for the current cluster
- Number of Pulsar-Brokers in the current cluster
- Number of Tenants in the current cluster

To cycle through the different status bar displays, press F1. The status bar display automatically refreshes at a configurable interval; to refresh the status bar display, press F5.

Apache Pulsar Deployment Representation

MSGMX displays the hierarchical structure of an Apache Pulsar deployment configuration using a file system-type representation to facilitate management of Apache Pulsar. The structure starts with instances at the top level and ends with namespaces and topics at the lower levels. The following is a representation of that structure:

instances / clusters / tenants / namespaces / topics

You can use the MSGMX `ls`, `pwd`, and `cd` commands to navigate this representation just as you would navigate a filesystem from the native command prompt. The prompt shows your location in the directory structure.

Command Overview

All MSGMX Apache Pulsar commands start with a verb that states the action to be taken — create, delete, list, etc. These verbs are listed in this manual in alphabetical order.

Commands have options or flags that can be required or optional. Options require an argument while flags do not. A required option or flag is presented as a selection from a list of choices. Optional ones are shown in this document enclosed in square brackets, "[]". Free or variable text entries are enclosed in angle brackets as shown here:

<FreeTextEntry>.

For example, to initiate a session with an Apache Pulsar deployment, instance Instance-A:

```
$ ./msgmx
> manage pulsar
> connect sample-instance localhost:8080
/sample-instance>
```

To navigate to Cluster-1:

```
/Instance-A> cd /Instance-A/Cluster-1
/Instance-A/Cluster-1>
```

You can also use relative paths, including "..", to navigate to a parent directory:

```
/Instance-A/Cluster-1/Tenant-A> cd Namespace-1/Topic-A
/Instance-A/Cluster-1/Tenant-A/Namespace-1/Topic-A> cd ..
/Instance-A/Cluster-1/Tenant-A/Namespace-1> cd ../..
/Instance-A/Cluster-1>
```

Also, you can use the `ls` command with full or relative pathnames, including "." to represent the current directory:

```
/Instance-A/Cluster-One/Tenant-A> ls /  
Instance-A  
/Instance-A/Cluster-One/Tenant-A> ls /Instance-A/Cluster-One  
Tenant-A  
Tenant-B  
/Instance-A/Cluster-One/Tenant-A> ls .  
Namespace-One  
Namespace-Two  
/Instance-A/Cluster-One/Tenant-A>
```

Command building is predictive, displaying applicable (and only applicable) options and flags that you can select with up/down arrows and the tab key to complete the word and advance to the next. Other standard MSGMX features also apply. See the MSGMX User Guide for details.



Note: For details on Apache Pulsar terminology, see pulsar.apache.org/docs/en/reference-terminology/. For more detailed explanations of options for command forms, see the [Apache Pulsar](#) documentation.

4lw (Apache ZooKeeper-specific)

Issue a ZooKeeper 4-letter-word command.

Use this command to access 4-letter-word ZooKeeper commands in a manner similar to issuing these commands via telnet directly to a ZooKeeper server.

By default 4-letter-word commands are disabled by ZooKeeper—you must enable them when configuring the ZooKeeper.

Example lines you can add to the `zookeeper.properties` file:

```
// Enable all 4lw commands  
4lw.commands.whitelist=*
```

```
// Enable only ruok, stat, conf, isro  
4lw.commands.whitelist=stat, ruok, conf, isro
```

Synopsis

4lw <command> [<host:port>]

Options

The following 4-letter-word ZooKeeper commands are supported by MSGMX.

- conf
- cons
- crst
- dump
- envi
- ruok
- srst
- svr

- stat
- wchc
- wchp
- wchs

For information on available Apache ZooKeeper 4-letter-word commands, see the [ZooKeeper Administrator's Guide, The Four Letter Words](#)

Examples

```
>4lw conf localhost:2181
```

```
/Instance/Cluster> 4lw srvr
```

cd

Change working directory.

Use this command to change to a different working directory within the Apache Pulsar instance hierarchy.

Synopsis

cd <path>

Command Form

cd directory_name

Navigate to a directory via full or relative path.

Examples:

```
/Instance/Cluster> cd some_tenant/some_namespace
/Instance/Cluster/some_tenant/some_namespace>

/Instance> cd /Instance/Cluster
/Instance/Cluster>

/Instance/Cluster> cd Tenant
/Instance/Cluster/Tenant>

/Instance/Cluster/Tenant> cd Namespace/Topic
/Instance/Cluster/Tenant/Namespace/Topic>
```

clear-backlog

Delete messages in a backlog.

Use this command to delete all backlogged messages for a specified namespace.

Synopsis

```
clear-backlog <namespace> [Namespace Options]
```

clear-backlog namespace

clear-backlog namespace <Namespace> [bundle <StartBoundry_EndBoundry>|subscription <SubscriptionName>]

Clear the message backlog for a specified namespace.

Examples:

The following command clears the backlog for NamespaceA within the specific bundle range:

```
/Instance> clear-backlog namespace /Instance/Cluster/Tenant/NamespaceA  
bundle 0x00000000_0x1234ffff
```

The following command clears the backlog for NamespaceA:

```
/Instance/Cluster/Tenant/NamespaceA> clear-backlog namespace .  
subscription sample-sub
```

The following command clears the backlog for the specific subscription using NamespaceA within the specific bundle range:

```
/Instance/Cluster/Tenant/NamespaceA> clear-backlog namespace .  
subscription sample-sub bundle 0x00000000_0x1234ffff
```

connect

Connect to an Apache Pulsar instance.

Use the connect command to associate your MSGMX session with an instance containing one or more Apache Pulsar Clusters. The target service must be running. An accessible IP address/port for the Apache Pulsar Broker's administrative interface must be supplied.

Synopsis

```
connect <InstanceName> <host:port>
    [allow-insecure-connection]
    [tls-certificate-file <filename>]
    [tls-key-file <filename>]
    [token <string>]
    [token-file <filename>]
    [trusted-certificate-file <filename>]
```

Required Arguments

InstanceName

The name you assigned when using the connect command is valid only while the connect session is active and provides a quick named way to access a specific cluster. InstanceName is shown as an instance at the root, or '/', level of the configuration hierarchy.

You can connect to multiple Apache Pulsar deployments and have multiple InstanceName active within a single MSGMX session by using multiple connect commands.

Optional Arguments

host:port

If needed, specify a host address of the Pulsar-Broker. This argument is needed only to connect to a new instance, not to instances that are already connected. It can be a URL specified with either `http://` or `https://`.

Example:

```
> connect Instance1 localhost:8080  
> connect existing_instance
```

allow-insecure-connection

This argument indicates that Messaging Manager must accept untrusted TLS certificates from the broker. The default setting is to not accept untrusted certificates.

tls-certificate-file

This argument specifies the client TLS certificate file that must be used to authenticate MTLS to the broker.

tls-key-file

This argument specifies the client TLS key file that is used to authenticate to the broker.

token

This argument is the token used to authenticate to the broker for token-based authentication schemes (OAuth2, JWT token, and so on).

token-file

This argument contains the file name that contains the authentication token. If both token and token-file arguments are specified, token takes precedence.

trusted-certificate-file

This argument specifies the file name of the trusted certificate that is used in the TLS communication.

See Also

[disconnect](#)

create

Create an object.

Use this command to create Apache Pulsar objects such as subscriptions, topics, namespaces, tenants, and schemas. You can create four different types of topics:

- persistent-topics
- persistent-partitioned-topics
- non-persistent-topics
- non-persistent-partitioned-topics



Note: As shown in the examples, the path can be absolute or relative to the current directory.

Synopsis

```
create <createCommandArg> [createCommandArg Options]
```

create namespace

```
create namespace <NamespaceName> [ bundle <Int>|clusters <Cluster>]
```

Create a new namespace.

Example:

Relative path:

```
/instance1/cluster1> create namespace tenant1/namespace1 bundle <int>  
clusters <Cluster>
```

create schemas

create schemas <Topic> file <FilePath>

Create a schema definition associated with a topic.

create tenant

create tenant <TenantName> [admin-roles <String>|allowed-clusters <Cluster>]

Create a new tenant. Separate multiple administrative roles or allowed clusters using a comma.

Example:

Absolute path:

```
/> create tenant /instance1/cluster1/tenant1 admin-roles <Roles>
allowed-clusters <Cluster>
```

create topic

create topic <Topic> [partitions <Number of Partitions>] [properties <Properties>]

Create a topic.

Examples:

Assume there is an instance "*instance1*", a cluster "*cluster1*", a tenant "*tenant1*", and a namespace "*namespace1*".

To create a topic, use the `create topic` command, followed by the topic path and name. To create a partitioned topic, add the `partitions` flag after the topic name, along with the number of partitions to use. To assign properties to the topic, add the `properties` flag, specifying properties as a comma-separated list of the form `Prop1=value,Prop2=value`.

The topic prefix determines whether the topic is persistent or non-persistent. The topic prefix can be left out, in which case the topic defaults to be created as a persistent topic. If included, the prefix must be either `persistent://` or `non-persistent://`. The prefix should go before the topic path.

Absolute path, persistent, non-partitioned topic:

```
/> create topic /instance1/cluster1/tenant1/namespace1/topic1

/> create topic
persistent://instance1/cluster1/tenant1/namespace1/topic1
```

Absolute path, non-persistent, non-partitioned topic:

```
/> create topic non-
persistent://instance1/cluster1/tenant1/namespace1/topic1
```

Absolute path, persistent, partitioned topic:

```
/> create topic

/instance1/cluster1/tenant1/namespace1/topic1 partitions 3

/> create topic
persistent://instance1/cluster1/tenant1/namespace1/topic1 partitions 10
```

Absolute path, non-persistent, partitioned topic:

```
/> create topic non-
persistent://instance1/cluster1/tenant1/namespace1/topic1 partitions 7
```

Relative path, persistent, non-partitioned topic:

```
/instance1/cluster1/tenant1> create topic persistent://namespace1/topic1
```

create subscription

create subscription <Subscription> message-id <Position> [properties <Properties>]

Create a subscription on a particular topic.

Examples:

Assume there is an instance "*instance1*", a cluster "*cluster1*", a tenant "*tenant1*", a namespace "*namespace1*", and a topic "*topic1*".

To create a subscription, use the `create subscription` command, followed by the subscription path and name. Specify the subscription's starting position in the message stream with the `message-id` flag (`earliest` starts the subscription at the earliest message in the topic, and `latest` starts the subscription at the latest message in the topic). To assign properties to the subscription, add the `properties` flag, specifying properties as a comma-separated list of the form `Prop1=value,Prop2=value`.

The topic prefix determines whether the topic is persistent or non-persistent. The topic prefix can be left out, in which case the topic defaults to a persistent topic. If included, the prefix must be either `persistent://` or `non-persistent://`. The prefix should go before the topic path.

Absolute path, persistent topic, starting at the earliest message:

```
/> create subscription
/instance1/cluster1/tenant1/namespace1/topic1/sub1 message-id earliest

/> create subscription
persistent://instance1/cluster1/tenant1/namespace1/topic1/sub1 message-
id earliest
```

Absolute path, non-persistent topic, starting at the latest message:

```
/> create subscription non-
persistent://instance1/cluster1/tenant1/namespace1/topic1/sub1 message-
id latest
```

Relative path, persistent topic, starting at the earliest message, and specifying subscription properties:

```
/instance1/cluster1/tenant1> create subscription
persistent://namespace1/topic1/sub1 message-id earliest properties
Prop1=A,Prop2=Z
```

create cluster

Create a new cluster definition.

The command enables you to make a cluster aware of another existing cluster. It does not actually create a cluster with running metadata stores, bookies, and brokers. It only creates the cluster definition to allow one cluster to communicate with another, existing cluster.

By default, the cluster definition is created within the current cluster.

Synopsis

```
create cluster <Cluster>
  [in <ExistingCluster>]
  [allow-insecure-connection true|false]
  [auth-plugin <Plugin>]
  [auth-parameters <Parameters>]
  [broker-client-certificate <Filename>]
  [broker-client-key <Filename>]
  [broker-client-tls true|false]
  [broker-client-tls-enable-keystore true|false]
  [broker-client-tls-keystore <Keystore>]
  [broker-client-tls-keystore-password <Password>]
  [broker-client-tls-keystore-type JKS|PKCS12]
  [broker-client-tls-truststore <Truststore>]
  [broker-client-tls-truststore-password <Password>]
  [broker-client-tls-truststore-type JKS|PKCS12]
  [broker-client-trust-certificate <Filename>]
  [broker-url <URL>]
  [broker-secure-url <URL>]
  [listener <String>]
  [proxy-protocol <SNI>]
  [proxy-url <URL>]
  [secure-url <URL>]
  [url <URL>]
```

The `in <ExistingCluster>` option allows you to create the cluster definition within a specific cluster.

The `broker-url` and `url` options are mandatory. Note that all `<Filename>` parameters are relative to the brokers of the created cluster, not to the host running TIBCO Messaging Manager.

delete

Delete an Apache Pulsar object.

Synopsis

```
delete <DeleteCommandArg> [DeleteCommandArg Options>]
```

delete bookie-rack

delete bookie-rack <Cluster> bookie <Bookie>

Remove the rack-placement information for a bookie.

delete cluster

delete cluster <Cluster> [in <Cluster>] [force]

Delete a cluster.

delete namespace

delete namespace <Namespace> [force]

Delete a namespace. The path can either be absolute, or relative to the current directory.

Example:

```
/instance1/cluster1/tenant1> delete namespace namespace1
```

delete schemas

delete schemas <Topic> [force]

Delete a schema definition associated with a topic.

delete subscription

delete subscription <Subscription> [force]

Delete a subscription. The path can be absolute or relative to the current directory.

Examples:

Deleting a subscription is essentially the same as creating one, except specifying the starting message is not required.

Absolute path, persistent topic:

```
/> delete subscription  
/instance1/cluster1/tenant1/namespace1/topic1/sub1  
  
/> delete subscription  
persistent://instance1/cluster1/tenant1/namespace1/topic1/sub1
```

Relative path, persistent topic from the topic directory:

```
/instance1/cluster1/tenant1/namespace1/topic1> delete subscription sub1
```

delete tenant

delete tenant <Tenant> [force]

Delete a tenant. The path can be absolute or relative to the current directory.

Example:

```
/> delete tenant /instance1/cluster1/tenant1
```

delete topic

delete topic <Topic> [force|delete-schema]

Delete a topic/partitioned-topic. Optionally you can also delete the schema associated with this topic. This single command handles partitioned or non-partitioned topics in the same manner. The path can be absolute or relative to the current directory.

Examples:

Deleting a topic is essentially the same as creating one, except specifying the partitions of a partitioned topic is not required.

Absolute path, persistent, non-partitioned topic:

```
/> delete topic /instance1/cluster1/tenant1/namespace1/topic1

/> delete topic
persistent://instance1/cluster1/tenant1/namespace1/topic1
```

Absolute path, non-persistent, non-partitioned topic:

```
/> delete topic non-
persistent://instance1/cluster1/tenant1/namespace1/topic1
```

Relative path, persistent, partitioned topic from the namespace directory:

```
/instance1/cluster1/tenant1/namespace1> delete topic topic1
```


disconnect

Disconnect MSGMX from a specific Apache Pulsar instance.

Use the `disconnect` command to disassociate your MSGMX session from a specific Apache Pulsar instance. If you have other connections active, and you disconnect from the instance you are currently `cd`'ed into, MSGMX changes back to the base directory, and you see the corresponding prompt:

```
/>
```

If you disconnect from an instance you are not `cd`'ed into, the prompt remains unchanged.

Synopsis

```
disconnect <InstanceName>
```

Required Arguments

InstanceName

The name assigned in the `connect` command.

Example:

```
> disconnect myInstance
```

See Also

[connect](#)

help

Display help information for any command.

Enter the `help` command or "?" to display information about a specific command.

Synopsis

`help <command-name>`

`? <command-name>`

`?` is an alias for `help`.

Required Options

Command name for which help is needed.

Examples:

```
Cluster@ZooKeeper:> help msgmx
Usage:
  msgmx source <File>
  msgmx load
  msgmx save
  msgmx set prediction < on|off >
  msgmx set summary < on|off >
  msgmx set timeout <Int>
  msgmx set status < on|off >
  msgmx set history-depth <Int>
  msgmx set echo-command < on|off >
  msgmx set color <ColorMode>
  msgmx set source-error <String>
  msgmx set cache-update-time <Int>
  msgmx set status-refresh <Int>
  msgmx set max-threads <Int>
  msgmx show all
  msgmx show prediction
  msgmx show summary

...Content truncated...
```

list

Display a list of Apache Pulsar objects.

Use this command to display a list of specified objects from the Apache Pulsar cluster to which you are connected.



Note: As shown in the examples, the path can either be absolute or relative to the current directory.

Synopsis

```
list <ListCommandArg> [ListCommandArg Options]
```

list bookies

list bookies

List bookies

list brokers

list brokers <Cluster> [dynamic-config]

List all Apache Brokers in the cluster to which you are currently connected.

list clusters

list clusters

List the existing clusters.

Example:

```
/instance1/cluster1> list clusters
```

list connections

list connections

List all current connections. This command also displays the current health status of the listed servers.

list failure-domains

list failure-domains [<Cluster>]

List the existing failure domains for a cluster.

list namespace-isolation-policies

list namespace-isolation-policies [<Cluster>]

List all namespace isolation policies of a specified cluster.

list namespaces

list namespaces [<Tenant>]

List the namespaces for a specified tenant.

list permissions

list permissions [<Topic>]

List the permissions on a specified topic.

list resource-groups

list resource-groups [<Cluster>]

List the resource groups on a specified cluster.

list subscriptions

list subscriptions [<Topic>]

List all subscriptions to a specified topic.

list tenants

list tenants [<Cluster>]

List the tenants in a cluster.

Examples:

```
/> list tenants /instance1/cluster1  
/instance1> list tenants cluster1
```

list topics

list topics <Namespace>

List specified topics.

See Also

[create](#), [delete](#)

ls

List the contents of a directory.

Use this command to display the contents of a directory.

Synopsis

```
ls [path]
```

Command Forms

ls

Display the contents of the current directory.

ls <Path>

Display the contents of the specified directory by absolute or relative path.

Examples

```
> ls /Instance
Cluster-A

/Instance/Cluster-A> ls
tenant-1
tenant-2

/Instance/Cluster-A> ls tenant-1
sample-namespace-a
sample-namespace-b

/Instance/Cluster-A/tenant-1> ls ../Cluster-A
tenant-1
tenant-2
```

pwd

Print working (current) directory.

Use this command to display the full path of the current directory.

Synopsis

pwd

Examples

```
/Instance/Cluster-A/tenant-1/sample-namespace-a> pwd  
/Instance/Cluster-A/tenant-1/sample-namespace-a
```

reset

Use this command to reset a specific object's value back to its default value on the Apache Pulsar cluster to which you are connected.

i Note: As shown in the examples, the path can either be absolute or relative to the current directory.

Synopsis

```
reset <ResetCommandArgs> [ResetCommandArg Options]
```

reset backlog-quota-check

reset backlog-quota-check

Manually trigger `backlogQuotaCheck`.

reset broker

reset broker <Cluster> <Broker Option>

Reset attributes for the specified broker.

The `reset broker` command has the following two forms.

reset broker <Cluster> dynamic-config <Config Parameter>

Reset the dynamic configuration parameter for the broker. The available dynamic configuration parameters can be viewed via the `show broker dynamic-config` command.

reset broker <Cluster> resource-quota <Namespace> bundle <StartBoundary_EndBoundary>

Reset the resource quota setting for the specified namespace and specified bundle in that namespace.

Examples:

```
/> reset broker localhost:8080 dynamic-config allowAutoTopicCreation

/> reset broker localhost:8080 resource-quota namespace
/instance1/cluster1/tenant1/namespace1 bundle 0x00000000_0x40000000
```

reset cluster

reset cluster <Cluster> [in <Cluster>] failure-domain <DomainName>

reset cluster <Cluster> [in <Cluster>] namespace-isolation-policy <Policy>

reset cluster <Cluster> [in <Cluster>] resource-group <GroupName>

reset cluster <Cluster> [in <Cluster>] peer-clusters

reset cluster <Cluster> [in <Cluster>] [allow-insecure-connection] [auth-plugin] [auth-parameters] [broker-client-certificate] [broker-client-key] [broker-client-tls] [broker-client-tls-enable-keystore] [broker-client-tls-keystore] [broker-client-tls-keystore-password] [broker-client-tls-keystore-type] [broker-client-tls-truststore] [broker-client-tls-truststore-password] [broker-client-tls-truststore-type] [broker-client-trust-certificate] [listener] [proxy-protocol <SNI>] [proxy-url <URL>]

Reset attributes for the specified cluster.

reset cluster <Cluster> failure-domain <DomainName>

Remove the named failure domain from the set of failure domains associated with the cluster.

Example:

```
/> reset cluster /instance1/cluster1 failure-domain Domain1
```

reset cluster <Cluster> namespace-isolation-policy <Policy>

Remove the named namespace isolation policy from the set of namespace isolation policies associated with the cluster.

Example:

```
/> reset cluster /instance1/cluster1 namespace-isolation-policy
NIPolicy1
```

reset cluster <Cluster> resource-group <GroupName>

Remove the named resource group from the set of resource groups associated with the cluster.

Example:

```
/> reset cluster /instance1/cluster1 resource-group Group1
```

reset namespace

reset namespace <NamespaceName> <Property>

Reset the value for a specified namespace property.

The `reset namespace` command requires one of the following property arguments. Some of these arguments require an additional identifier argument. For details, refer to the Apache Pulsar documentation.

- anti-affinity-group
- auto-subscription-creation
- auto-topic-creation
- backlog-quotas with options:
 - type
- bookie-affinity-group
- compaction-threshold
- deduplication
- delayed-delivery
- dispatch-rate
- inactive-topic-policies
- max-consumers-per-subscription

- max-consumers-per-topic
- max-producers-per-topic
- max-subscriptions-per-topic
- max-topics
- max-unacked-messages-per-consumer
- max-unacked-messages-per-subscription
- message-ttl
- offload-deletion-lag
- offload-policies
- permissions <Role>
- persistence
- properties
- publish-rate
- replicator-dispatch-rate
- resource-group
- retention
- subscribe-rate
- subscription-dispatch-rate
- subscription-expiration-time
- subscription-permissions <Subscription> roles <Role>
- subscription-types-enabled

Examples:

```
/> reset namespace /instance1/cluster1/tenant1/namespace1 permissions  
admin-15
```

```
/> reset namespace /instance1/cluster1/tenant1/namespace1 subscribe-rate
```

```
/instance1/cluster1/tenant1/namespace1> reset namespace . backlog-quotas
```

reset subscription

reset subscription <Subscription> earliest

reset subscription <Subscription> latest

reset subscription <Subscription> message-id <MessageID>

Reset the named subscription's message position.

reset subscription <Subscription> earliest

Position the named subscription to the earliest message in the topic.

reset subscription <Subscription> latest

Position the named subscription to the latest message in the topic.

reset subscription <Subscription> message-id <MessageID>

Position the named subscription to the specific message ID in the topic.

Example:

```
/> reset subscription  
persistent://instance1/cluster1/tenant1/namespace1/topic1/sub1 message-  
id 203:0
```

reset topic

reset topic <Topic> permissions <Role>

reset topic <Topic> truncate

Reset topic attributes.

reset topic <Topic> permissions <Role>

Reset permissions for a specified role in a topic.

Example:

```
/> reset topic /instance1/cluster1/tenant1/namespace1/topic1 permissions  
admin-15
```

reset topic <Topic> truncate

Truncate the named topic.

See Also

[set](#)

set

Use this command to set an object's configuration value on the Apache Pulsar cluster to which you are connected.

i Note: As shown in the examples, the path can either be absolute or relative to the current directory.

Synopsis

```
set <SetCommandArg> [SetCommandArg Options]
```

set bookie

```
set bookie-rack <Cluster> bookie <Bookie> rack-name <RackName> [group <Bookie Group Name>|hostname <Bookie Host Name>]
```

Configure a bookie rack.

set broker

```
set broker <Cluster> dynamic-config <ConfigParameterName> <ConfigValue>
```

```
set broker <Cluster> resource-quota msg-rate-in <Value> msg-rate-out <Value> memory <Value> bandwidth-out <Value> bandwidth-in <Value> [namespace <Instance/Cluster/Tenant/Namespace> bundle <StartBoundry_EndBoundry>] dynamic <true/false>
```

Set values for a broker.

```
set broker <Cluster> dynamic-config <ConfigParameterName> <ConfigValue>
```

Set the value of a dynamic configuration parameter for a broker. The available dynamic configuration parameters can be viewed via the `show broker dynamic-config` command.

Example:

```
/> set broker localhost:8080 dynamic-config allowAutoTopicCreation false
```

set broker <Cluster> resource-quota msg-rate-in <Value> msg-rate-out <Value> memory <Value> bandwidth-out <Value> bandwidth-in <Value> [namespace <Instance/Cluster/Tenant/Namespace> bundle <StartBoundry_EndBoundry>] [dynamic <true/false>]

Set the resource quota for a broker.

Examples:

Set the resource quota to 3000 messages per second in and out, 128MB of memory, and bandwidth in and out to 10000 bytes per second.

```
/> set broker localhost:8080 resource-quota msg-rate-in 3000 msg-rate-out 3000 memory 128 bandwidth-out 10000 bandwidth-in 10000
```

Set the resource quota to 2000 messages per second in and out, 128MB of memory, bandwidth in and out to 10000 bytes per second, and allow the rates to be dynamically recalculated.

```
/> set broker localhost:8080 resource-quota msg-rate-in 2000 msg-rate-out 2000 memory 128 bandwidth-out 10000 bandwidth-in 10000 dynamic true
```

Set the resource quota to 5000 messages per second in and out, 128MB of memory, bandwidth in and out to 15000 bytes per second, and allow the rates to be dynamically recalculated. Apply the resource quota to bundle 0x00000000_0x40000000 of namespace /instance1/cluster1/tenant1/namespace1.

```
/> set broker localhost:8080 resource-quota msg-rate-in 5000 msg-rate-out 5000 memory 128 bandwidth-out 15000 bandwidth-in 15000 namespace /instance1/cluster1/tenant1/namespace1 bundle 0x00000000_0x40000000
```

set cluster

set cluster <Cluster> [in <Cluster>] failure-domain <Domain> [broker <BrokerList>]

```
set cluster <Cluster> [in <Cluster>] namespace-isolation-policy <Policy> type
<failoverPolicyType> min-limit <Integer> usage-threshold <Integer> namespaces
<namespace-list> primary <BrokerList> [secondary <BrokerList>]
```

```
set cluster <Cluster> [in <Cluster>] peer-cluster <ClusterList>
```

```
set cluster <Cluster> [in <Cluster>] resource-group <GroupName> [dispatch-bytes
<Integer>] [dispatch-messages <Integer>] [publish-bytes <Integer>] [publish-messages
<Integer>]
```

```
set cluster <Cluster> [in <Cluster>] [allow-insecure-connection true|false] [auth-plugin
<Plugin>] [auth-parameters <Parameters>] [broker-client-certificate <Filename>]
[broker-client-key <Filename>] [broker-client-tls true|false] [broker-client-tls-enable-
keystore true|false] [broker-client-tls-keystore <Keystore>] [broker-client-tls-
keystore-password <Password>] [broker-client-tls-keystore-type JKS|PKCS12] [broker-
client-tls-truststore <Truststore>] [broker-client-tls-truststore-password <Password>]
[broker-client-tls-truststore-type JKS|PKCS12] [broker-client-trust-certificate
<Filename>] [broker-url <URL>] [broker-secure-url <URL>] [listener <String>] [proxy-
protocol <SNI>] [proxy-url <URL>] [secure-url <URL>] [url <URL>]
```



Note: All the <Filename> parameters are relative to the cluster broker and not to the host running Messaging Manager.

Set values for a specified cluster.

```
set cluster <Cluster> failure-domain <DomainName> [broker <BrokerList>]
```

Create or modify a failure domain for the cluster.

The optional <BrokerList> is a comma-separated list of broker host:port entries.

Examples:

```
/> set cluster /instance1/cluster1 failure-domain domain1

/> set cluster /instance1/cluster1 failure-domain domain1 broker
broker1:8623,broker2:8723
```

```
set cluster <Cluster> namespace-isolation-policy <Name> type <FailoverPolicyType>
min-limit <Int> usage-threshold <Int> namespaces <Namespaces> primary <Brokers>
```


[secondary <Brokers>]]

Create or modify a namespace isolation policy for the cluster.

- <Namespaces> is a comma-separated list of plain namespace names, e.g. not including the `/instance/cluster/` prefix.
- <Brokers> is a comma-separated list of broker host:port entries.
- Pulsar currently only supports the <FailoverPolicyType> value of `min_available`.

Examples:

```
set cluster /instance1/cluster1 namespace-isolation-policy policy1 type
min_available min-limit 300 usage-threshold 600 namespaces
namespace1,namespace2 primary localhost:8080

set cluster /instance1/cluster1 namespace-isolation-policy policy1 type
min_available min-limit 300 usage-threshold 600 namespaces
namespace1,namespace2 primary broker1:8623 secondary broker2:8723
```

set cluster <Cluster> peer-clusters <Clusters>

Create or modify a cluster's list of peer clusters.

- <Clusters> is a comma-separated list of cluster paths, e.g. `/instance/cluster`

Example:

```
set cluster /instance1/cluster1 clusters
/instance1/cluster2,/instance1/cluster3
```

set cluster <Cluster> resource-group <GroupName> [dispatch-bytes <Int>] [dispatch-messages <Int>] [publish-bytes <Int>] [publish-messages <Int>]

Create or modify a cluster's resource group.

Example:

```
set cluster /instance1/cluster1 resource-group rg1 dispatch-bytes 10000
publish-bytes 10000
```

set namespace

set namespace <NameSpaceName> <Namespace Option> (See [set](#) which follows.)

Set values for the specified namespace.

Examples:

```
/> set namespace /instance1/cluster1/tenant1/namespace1 permissions
admin-03 actions produce

/> set namespace /instance1/cluster1/tenant1/namespace1 message-ttl 1000

/instance1/cluster1/tenant1/namespace1> set namespace . replication-
cluster /instance1/cluster2

/instance1/cluster1/tenant1/namespace1> set namespace . deduplication
enable
```

Set Namespace Options

For details on the following options, refer to the [Apache Pulsar documentation](#).

- anti-affinity-group <AntiAffinityName>
- auto-subscription-creation <enable|disable>
- auto-topic-creation type non-partitioned [create <enable|disable>]
- auto-topic-creation type partitioned num-partitions <Int> [create <enable|disable>]
- auto-update-schema <enable|disable>
- backlog-quota <SizeLimit> policy <PolicyName> [type <QuotaType> limit_time <Int>]
 - Policy Name Options:
 - producer_request_hold
 - producer_exception

- consumer_backlog_eviction
- Quota Types:
 - destination_storage
 - message_age
- bookie-affinity-group primary <Group Name> secondary <Group Name>
- compaction-threshold <Size>
- deduplication <enable|disable>
- deduplication-snapshot-interval <Seconds>
- delayed-delivery [delay <enable|disable>] [time <Seconds>]
- dispatch-rate [bytes <Int>] [seconds <Int>] [messages <Int>] [relative-to-publish-rate <true|false>]
- encryption-required <enable|disable>
- inactive-topic-policies [delete <enable|disable>] [max-inactive-duration <Seconds>] [mode <Mode>]
- Modes:
 - delete_when_no_subscriptions
 - delete_when_subscriptions_caught_up']
- max-consumers-per-subscription <Int>
- max-consumers-per-topic <Int>
- max-producers-per-topic <Int>
- max-subscriptions-per-topic <Int>
- max-topics <Intd>
- max-unacked-messages-per-consumer <Intd>
- max-unacked-messages-per-subscription <Intd>
- message-ttl <Seconds>
- offload-deletion-lag <Size>

- `offload-policies driver <Driver> bucket <BucketName> [aws-id <String>] [aws-secret <String>] [directory <DirectoryPath>] [endpoint <String>] [file-system-uri <URI>] [file-system-profile-path <Path>] [max-block-size <Int>] [offload-after-elapsed <Int>] [offload-after-threshold <Int>] [offloaded-read-priority <Read Priority>] [read-buffer-size <Int>] [region <String>] [s3-role <RoleName>] [s3-session-name <SessionName>]`

- Read Priority values:

- `TIERED_STORAGE_FIRST`
- `BOOKKEEPER_FIRST`

- `offload-threshold <Size>`
- `permissions <Role> actions <Actions>`

Actions is a comma-separated list containing one or more of the following values:

- `produce`
- `consume`
- `sources`
- `sinks`
- `functions`
- `packages`

- `persistence persistence [bookkeeper-ack-quorum <Int>] [bookkeeper-ensemble <Int>] [bookkeeper-write-quorum <Int> ml-mark-delete-max-rate <Float>]`
- `properties <PropertyList>`

PropertyList is a comma-separated list of items, each item of the form `<Name>=<Value>`.

- `publish-rate [byte-publish-rate <Int>] [msg-publish-rate <Int>]`
- `replication-cluster <Cluster>`
- `replicator-dispatch-rate [bytes <Int>] [seconds <Int>] [messages <Int>] [relative-to-publish-rate <true|false>]`
- `resource-group <GroupName>`

- retention size <Megabytes> time <Minutes>
- schema-compatibility-strategy <Compatibility>

Compatibility is one of the values:

- FULL
 - BACKWARD
 - FORWARD
 - UNDEFINED
 - BACKWARD_TRANSITIVE
 - FORWARD_TRANSITIVE
 - FULL_TRANSITIVE
 - ALWAYS_COMPATIBLE
 - ALWAYS_INCOMPATIBLE
- schema-validation-enforce <enable|disable>
 - split-bundle <StartBoundry_EndBoundry> [unload <Value>]
 - subscribe-rate [rate-per-consumer <Int>] [rate-period-seconds <Int>]
 - subscription-auth-mode <None|Prefix>
 - subscription-dispatch-rate [bytes <Int>] [seconds <Int>] [messages <Int>] [relative-to-publish-rate <true|false>]
 - subscription-expriation-time <Minutes>
 - subscription-permissions <Subscription> role <Roles>
 - subscription-types-enabled <Subscription Type>
 - Exclusive
 - Shared
 - Failover
 - Key_Shared
 - unsubscribe <SubscriptionName> [bundle <Bundle>]

set subscription

set subscription <Subscription> <Subscription Option>

Set attributes for a specified subscription.

Options include:

- expire-messages [age <Seconds>]
- replicated-subscription-status <enable/disable>
- skip all
- skip count <Int>

Examples:

```
/> set subscription /instance1/cluster1/tenant1/namespace1/topic1/sub1
expire-message

/> set subscription /instance1/cluster1/tenant1/namespace1/topic1/sub1
skip count 10

/instance1/cluster1/tenant1/namespace1/topic1> set subscription sub1
skip all

/instance1/cluster1/tenant1/namespace1/topic1> set subscriptions sub1
replicated-subscription-status enable
```

set tenant

set tenant <Tenant> [admin-roles <AdminRole>] [allowed-clusters <Clusters>]

Assign role and cluster to a tenant.

Examples:

```
/> set tenant /instance1/cluster1/tenant1 admin-role admin-12

/instance1/cluster1> set tenant . cluster /instance1/cluster2
```

set topic

set topic <Topic> <Topic Option>

Set values for a specified topic.

Options include:

- compact
- expire-messages age <Seconds>
- partitions <Int>
- permissions <Actions> role <Role>
- replication-clusters <Cluster1,Cluster2>
- terminate

Examples:

```
/> set topic /instance1/cluster1/tenant1/namespace1/topic1 permissions  
produce role admin-21  
  
> set topic /instance1/cluster1/tenant1/namespace1/topic1 partitions 9  
  
/instance1/cluster1/tenant1/namespace1/topic1> set topic . terminate  
  
/instance1/cluster1/tenant1/namespace1/topic1> set topic . expire-  
messages age 1200
```

See Also

[reset](#)

show

Use this command to display an object's configuration values on the Apache Pulsar cluster to which you are connected.



Note: As shown in the examples, the path can either be absolute or relative to the current directory.

Synopsis

```
show <ShowCommandArg> [ShowCommandArg Options]
```

show bookie-rack

```
show <Cluster> [bookie <Bookie>] bookie-rack <Broker Host:Port> [bookie <Bookie Name>]
```

Show information about a bookie.

show broker

```
show broker <Cluster> [<broker arguments and options>]
```

Show information about brokers in the cluster.

Options include:

- `allocator-stats <AllocatorName>`
- `bookieops`
- `dynamic-config`
- `health [topic-version <V1/V2>]`
- `internal-config`

- leader-broker
- load-report
- mbeans
- monitoring-metrics
- resource-quota [namespace <Namespace> bundle <Bundle>]
- runtime-config
- topic-stats
- version

Examples:

```
/> show broker http://127.0.0.1:8080 internal-config
/> show broker http://127.0.0.1:8080 dynamic-config
```

show cluster

show cluster <Cluster> [in <Cluster>]

show cluster <Cluster> [in <Cluster>] failure-domain <DomainName>

show cluster <Cluster> [in <Cluster>] namespace-isolation-policies [brokers | broker <Broker>]

show cluster <Cluster> [in <Cluster>] namespace-isolation-policy <Policy>

show cluster <Cluster> [in <Cluster>] peer-clusters

show cluster <Cluster> [in <Cluster>] resource-group <GroupName>

Show information about a cluster.

Examples:

```
/instance1/cluster1> show cluster . peer-cluster
```

show namespace

show namespace <Namespace> [<namespace options>]

Show information about a specified namespace.

show namespace Options

For details on the following options, refer to the [Apache Pulsar documentation](#).

- anti-affinity-group
- auto-subscription-creation
- auto-topic-creation
- auto-update-schema
- backlog-quota
- bookie-affinity-group
- bundles
- compaction-threshold
- deduplication
- deduplication-snapshot-interval
- delayed-delivery
- dispatch-rate
- encryption-required
- inactive-topic-policies
- max-consumers-per-subscription
- max-consumers-per-topic
- max-producers-per-topic
- max-subscriptions-per-topic
- max-topics

- max-unacked-messages-per-consumer
- max-unacked-messages-per-subscription
- message-ttl
- offload-deletion-lag
- offload-policies
- offload-threshold
- permissions
- persistence
- policies
- properties
- publish-rate
- replication-cluster
- replicator-dispatch-rate
- resource-group
- retention
- schema-compatibility-strategy
- schema-validation-enforce
- subscribe-rate
- subscription-auth-mode
- subscription-dispatch-rate
- subscription-expiration-time
- subscription-permissions
- subscription-types-enabled

Examples:

```
/> show namespace /instance1/cluster1/tenant1/namespace1 permissions
```

```
/instance1/cluster1/tenant1/namespace1> show namespace . policies
```

show racks-placement

show racks-placement <Cluster>

Show information about a bookie rack.

Example:

```
/> show racks-placement http://127.0.0.1:8080
```

show schemas

show schemas <Topic> [version <Version>]

Show information about schemas based on a specified topic in the cluster.

Examples:

```
/> show schemas /instance1/cluster1/tenant1/namespace1/topic1  
/> show schemas /instance1/cluster1/tenant1/namespace1/topic1 version 03  
/instance1/cluster1/tenant1/namespace1/topic1> show schemas . version 06
```

show tenant

show tenant [<Tenant>]

Show information about a tenant.

Example:

```
/instance1/cluster1/tenant1> show tenant .
```

show topic

show topic <Topic> [< topic arguments and options>]

Show information about a specified topic in the cluster.

Options include:

- backlog-size
- bundle-range
- compact-status
- last-message-id
- lookup
- message [count <Int>] [position <Int>] [start <earliest|latest>]
- message id <MessageID>
- message-id <DateTime>
- offload-status
- partitioned-metadata
- properties
- replication-clusters
- stats [info-internal | internal]

Examples:

```
/> show topic /instance1/cluster1/tenant1/namespace1/topic1 lookup  
/> show topic /instance1/cluster1/tenant1/namespace1/topic1 bundle-range  
/>/instance1/cluster1/tenant1/namespace1/topic1> show topic . stats  
/instance1/cluster1/tenant1/namespace1/topic1> show topic . last-  
message-id
```

unload

Unload a topic or namespace.

Use this command to manually unload topics or namespaces from a broker and reassign the topics to a new broker, based on current load.

i Note: As shown in the examples, the path can either be absolute or relative to the current directory.

Synopsis

```
unload <unloadCommandArg> [unloadCommandArg Options]
```

unload namespace

unload namespace <Namespace> [bundle <StartBoundry...EndBoundry>]

Unload topics for a namespace and trigger reassignments.

Examples:

```
/instance1/cluster1/tenant1/namespace1> unload namespace .  
  
/instance1/cluster1/tenant1/namespace1> unload namespace . bundle  
0x000000_0xffffffff
```

unload topic

unload topic <Topic>

Close the topics, release ownership and reassign the topics to a new broker, based on current load.

Example:

```
/> unload topic /instance1/cluster1/tenant1/namespace1/topic1
```

TIBCO Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the [Product Documentation website](#), mainly in HTML and PDF formats.

The [Product Documentation website](#) is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

Documentation for this product is available on the [TIBCO® Messaging Manager Product Documentation](#) page:

Updated Resources on TIBCO Community

Supplemental resources are now distributed at the [TIBCO Messaging Community Wiki](#) in the Reference Info tab. You can always find the latest versions of these resources in that location.

Those resources include quick start guides, tutorials, and usage examples.

How to Contact Support for TIBCO Products

You can contact the Support team in the following ways:

- To access the Support Knowledge Base and getting personalized content about products you are interested in, visit our [product Support website](#).
- To create a Support case, you must have a valid maintenance or support contract with a Cloud Software Group entity. You also need a username and password to log in to the [product Support website](#). If you do not have a username, you can request

one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the [TIBCO Ideas Portal](#). For a free registration, go to [TIBCO Community](#).

Legal and Third-Party Notices

SOME CLOUD SOFTWARE GROUP, INC. (“CLOUD SG”) SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, “INCLUDED SOFTWARE”). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

USE OF CLOUD SG SOFTWARE AND CLOUD SERVICES IS SUBJECT TO THE TERMS AND CONDITIONS OF AN AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER AGREEMENT WHICH IS DISPLAYED WHEN ACCESSING, DOWNLOADING, OR INSTALLING THE SOFTWARE OR CLOUD SERVICES (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH LICENSE AGREEMENT OR CLICKWRAP END USER AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE SAME TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc.

TIBCO, the TIBCO logo, the TIBCO O logo, TIB, Information Bus, TIB, FTL, eFTL, and TIBCO Enterprise Message Service are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG’s Third Party Trademark Notices (<https://www.cloud.com/legal>) for more information.

This document includes fonts that are licensed under the SIL Open Font License, Version 1.1, which is available at: <https://scripts.sil.org/OFL>

Copyright (c) Paul D. Hunt, with Reserved Font Name Source Sans Pro and Source Code Pro.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file

for the availability of a specific version of Cloud SG software on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "README" FILES.

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at <https://www.tibco.com/patents>.

Copyright © 2019-2025. Cloud Software Group, Inc. All Rights Reserved.