Tags

Summary

Guidance and examples for using tags in instructions.

On this page:

- Tag Types
  - Device
  - Freeform
- Tag Storage
- Tag Methods
- Defining Device Tags

Tag Types

Tachyon supports two kinds of tag that can be used to set properties on Tachyon Agent devices. The Tachyon Agent stores both types of tag in persistent storage. Values are set or deleted using actions and queried using questions. The differences are described below.

Tag type	Description
Device	Also known as **scopable** tags in Agent methods documentation, and previously known as **Coverage** tags in earlier versions of Tachyon - see info panel below. Can also be used when defining the coverage for questions and actions. A Tachyon **Custom Properties Administrator** must define the names and values for Device tags before they can be applied to devices using Device tag actions. See Defining Device Tags below. Once set by an action, the Tachyon Agent reports Device tags back to the System where they are stored in the Tachyon Master database for each device (in the `Device` table), which is how they can be used to define coverage for questions and actions. Instructions which query tag contents, do so by querying the Agent, they do not query the Master database; the database is used only for coverage. The Tachyon Agent regularly reports back its list of tags. The list consists of a separator and **name=value** for each tag plus a final separator at the end of the list. The entire list of **name=value** pairs and their separators must not exceed 511 bytes. Any tags that exceed this limit are not reported back. Only the tags reported back by each Tachyon Agent can be used for coverage. Therefore, the length of tag names and values should be kept short if many different tags are to be used. The maximum length of a tag name is 16 characters, and the maximum length of a tag value is 32.
Freeform	Also known as **non-scopable** tags in Agent methods documentation. Cannot be used when defining the coverage for a question. Freeform tags can be set with arbitrary names and values that are defined only when running a Freeform tag action. Freeform tags exist purely on the Agent, where they can be set, their values fetched, and checked for existence. The list of freeform tags consists of a separator and **name=value** for each tag, plus a final separator at the end of the list. There is no limit to the number of freeform tags or the length of the list, and no limit to the length of tag name or tag value.

Tag names are case-insensitive. Tag values are not case-sensitive, and case is preserved.

Prior to 8.0 Device tags were called Coverage tags. Device tags behave exactly the same as Coverage tags.

Only the name has been changed, to help distinguish the difference with Software tags. Device tags are *client-side* and managed using SCALE and used for coverage in Explorer. Software tags are *server-side* and used in Software inventory.

From 8.0 onwards, both Device and Software tags can be used in Management Group rules, provided they are defined in Settings Custom Properties.

In 1E Client 8.0, a new Windows installer property TAGS has been introduced, to support setting of Device and Freeform tags during installation. Please refer to Tachyon client settings.

Tag Storage

Tags are stored on devices in Agent persistent storage, and as such persist during re-installation and upgrades. They are deleted only if explicitly deleted or the persistent storage is deleted.

Tag names are alphanumeric strings, which are case-insensitive but returned in upper-case.

Tag values are alphanumeric strings, with case preserved. That is, a number would be represented as a string such as "99", and can be empty (zero-length string) if the presence or absence of a tag is sufficient.

Device and freeform tags are stored internally as two distinct lists.

Each list uses | as a separator between each NAME=Value and at the start and end of the list. For example:
|DEPT=Sales|LOCATION=London|COUNTRY=UK|REGION=EMEA|

|| is an empty tag list.

A tag name must be unique within a list, so that assigning a value to a tag will create the tag if it does not already exist or overwrite the tag's value if it was already there.

Device and freeform tags can share the same names and they are distinct.

Tag Methods

The following methods existing in the Agent Tagging module:

- Tagging.Check — Tests the existence of a named tag of a specified type, optionally with the specified value.
- Tagging.Clear — Delete all tags of the specified type.
- Tagging.Count — Get the quantity of tags of the specified type.
- Tagging.Delete — Delete the named tag of the specified type, and indicate if it originally existed.
- Tagging.Get — Indicate whether a named tag of the specified type is present and return its value if present.
- Tagging.GetAll — Fetch the tag list (containing all tags) of the specified type.
- Tagging.Set — Set (or change if already set) the named tag of the specified type to the specified value.

Each method requires the tag type to be specified as either scopable (`true`) or non-scopable (`false`), with default scopable (`true`).

When setting a tag, you will need to cast a non-text value as text.

Example

```
@toset = 99;
@toset = SELECT CAST(Value AS text) AS Value FROM @toset;
Tagging.Set(Name: "EXAMPLETAG", Value: @toset.Value, Scopable: false);
```

Defining Device Tags

Users of the Tachyon Explorer UI can target specific devices based on various attributes, including Device tags. A Device tag may only be used to target devices if the tag's name and allowed values are already defined in the Tachyon system **Custom Properties** admin page. Instructions can then be configured to prompt users for selection of defined name and values. Whilst it is possible to set *other* Device tag names and values on Agents, it is best not to do this because they cannot be used for coverage until their names and values are defined in **Custom Properties** admin page. More details on the Device tag workflow and tutorials for administrators are given in Tachyon version documentation, for example using Tachyon 5.1 see: Tagging client devices.

Device Tag is a type of custom property, and is currently the only type of custom property that is configurable in the Custom Properties admin page. A Tachyon **Custom Properties Administrator** must define the names and values for Device tags before they can be applied to devices using Device tag actions.

It is possible for a Device tag to exist on Agents which do not exist as a Custom Property. This may occur because an instruction has set a Device tag without reference to Custom Properties, or a Custom Property has been deleted from the system after being set on Agents. It is good practice to define custom properties in the admin page, and then use the instruction **Set Device tag %tagname% to %tagvalue%** which uses "value picker" parameters that let you select a tag name and value based on defined custom properties. If you set a Device tag on an Agent that does not exist as a custom property, then it will behave like a freeform tag and you will not be able to use it to set coverage in the Tachyon Explorer. Furthermore, you will use up space in the 511 character length of the Device tag list.

See the following links in the SDK for details about value picker parameters.

- Instruction Definition Reference (ParameterJson)
- Getting Started (Parameter properties)

While on the subject of Device tags that don't exist as defined custom properties. Consider a hypothetical situation where Device tags have been defined, and set on many Agents, and then the Tachyon server hardware fails and the Master database is lost. If the server is rebuilt using a new database, then you would probably want to re-instate the same Device tags. If you did not have these documented, then you could use the `GetAll` method (see Tag Methods) to query all devices for details of all Device tag names, and use this to re-define Device tags. Alternatively you could start over again by establishing new tags, and using the `Clear` method to delete existing tags on devices.