Table of Contents

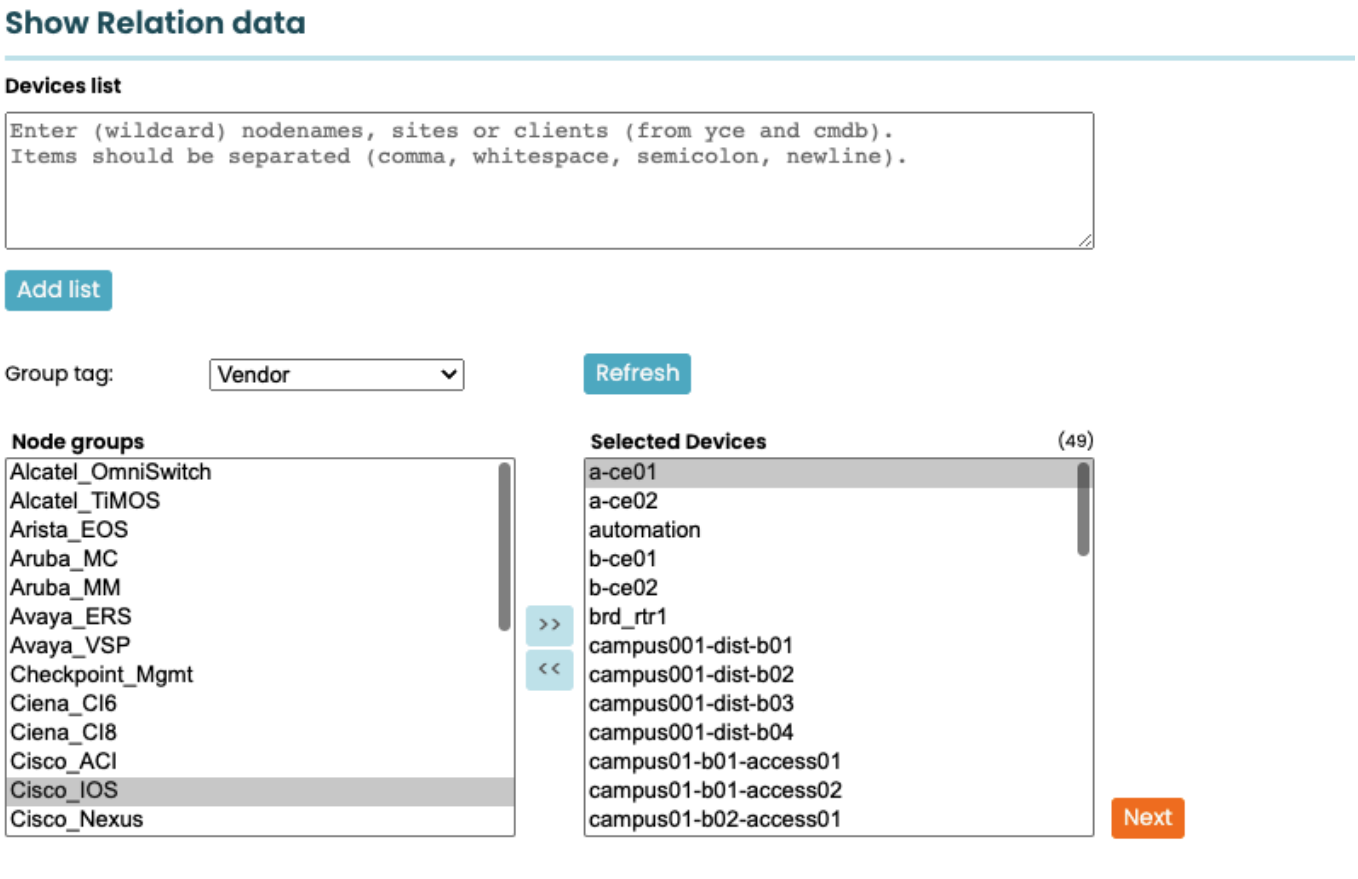
Relation test

The relation test tool enables you to execute the queries made in the Relations form for any node, so you are able to review the resulting data.

First page

This tool uses two pages, the first to select the nodes (devices) to test the relation(s) with, the second to find and select the relation and evaluate the results for the context (the node).

Devices can be selected by several means. Adding a Node-group will expand the 'Selected devices' box with the nodes matching the group criteria and the permissions the user has for the client-type of each node (minimally 'engineer'). Selecting more Node-groups (by double-clicking or hitting **>>**) will further expand the 'Selected devices' box.



Alternatively devices can be added by typing a list of Node names, ClientCodes or SiteCode in the 'Devices list' box and clicking the 'Add list' button. Each entry will be sequentially be attempted to expand into one or more device names. What is more, these list items support the wildcards * and **?** to quickly find ranges of devices.

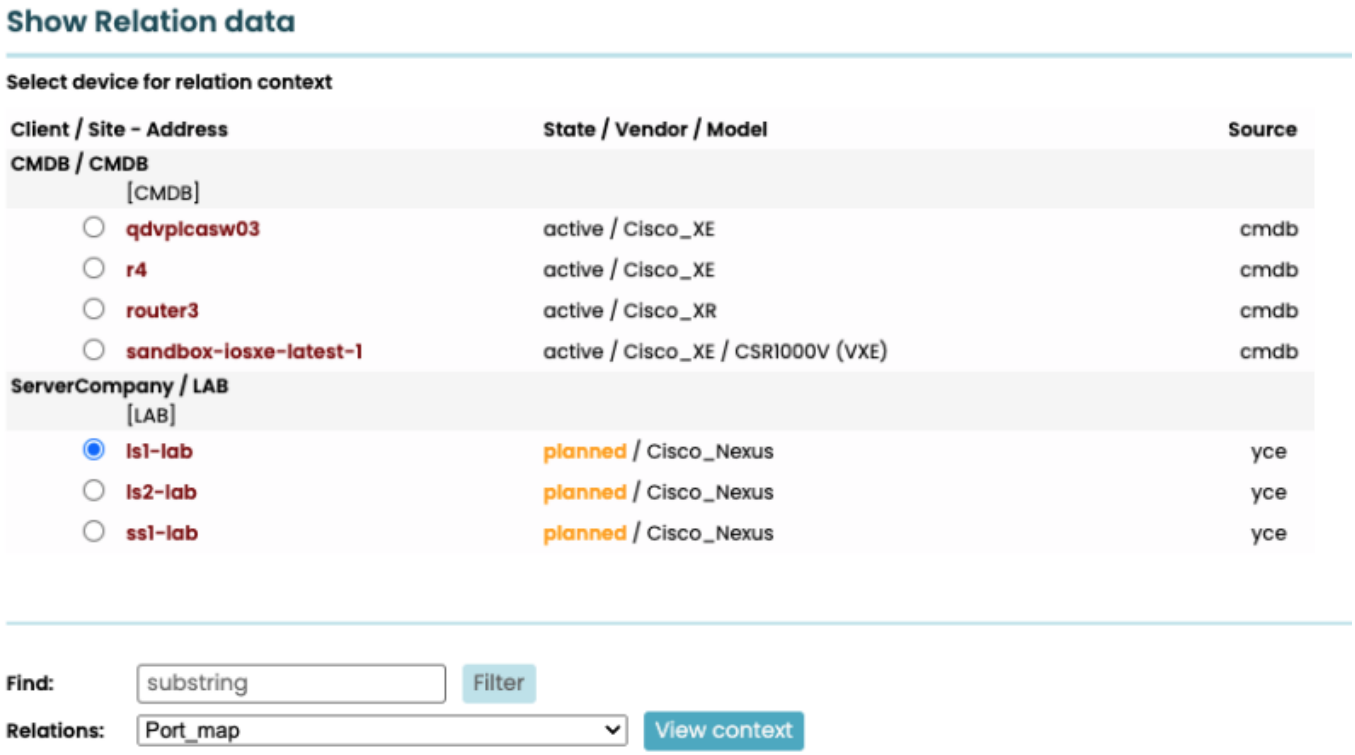
Selected Devices can be found in the YCE database or in the CMDB database. Where both exist, the YCE database takes precedence as its context is much more extensive. Its origin will be displayed in the column 'Source' of the second page.

Multiple nodes can be selected in the 'Selected devices' box and removed from the list by clicking the **<<** button.

Second page

The second screen is displayed after clicking **`Next`**

This page has three sections. At the top is a list of all selected devices grouped per Client and Site along with the location address. For each node the nodename, its State, Vendor-type, Model and its Source are given. The Model will only be available once NetYCE executed a job on the device to retrieve its device model.



One of the nodes can be selected to become the 'current context', that is, the collection of database information related to the selected node, the values used for retrieving the relation data.

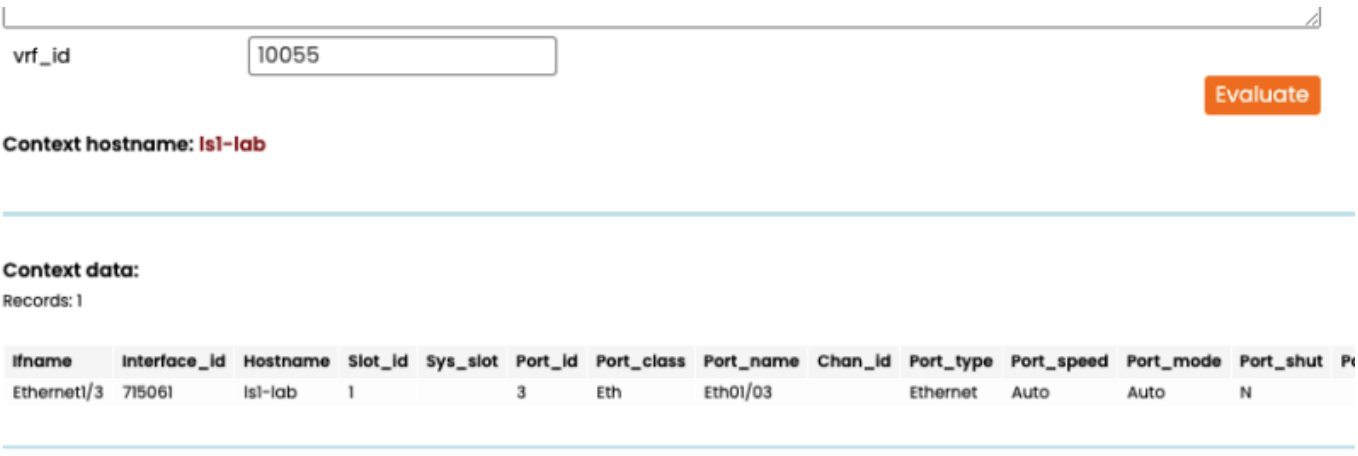
Next the section for selecting the relation to test is shown. As Relations are global (no client-type restrictions apply), a filter can be used to reduce the number of offered relations. Select the one to test which query is then retrieved and can be edited.

> Note: edited relations are *not* saved. Relations must be changed using the Relations form.

In the example above, the Relation uses a variable in its query that cannot be substituted directly from the nodes context and therefore prevents the query to retrieve data. The orange “Warning” informs that the Parameter <vlan> could not be resolved.

When examining the relation query, the parameters present in the query were included in the form to allow the user to provide an overriding value. The example also shows the <client_Type> parameter, but it could be substituted or would also have issued a warning.

So, after entering a sensible value for <vlan> the query can be executed again by clicking the Evaluate button and the resulting records (if any) will be displayed.



Most Relations will execute properly without providing additional values as their scope used the node's context. But especially for relations used in vlan- or interface-templates, providing a manual value for vlan id or port name is essential.

Not all parameters used in the relation query are available for an overriding value: Hostname, ClientCode and SiteCode are suppressed as they are already manually selected in the device list.

YCE vs CMDB nodes

Although the Relation test tool can operate on both YCE and CMDB nodes, it should be understood that the information and therefore the context of a CMDB node is very limited. A CMDB node has only a limited set of attributes and only links to YCE objects like Domain, Region, Client and Site. It currently lacks interfaces, services and subnets required to create complex relations.

As a result, CMDB nodes will likely require its own set of Relations, tuned to the available context.