Service-types

Service-types are basically scripted tasks which perform an automated task on the network in the database. These tasks can be:

1. Add/remove devices to the network
2. Add/remove topology
3. Add/remove VLAN/IP-addresses
4. Add/remove/enable/disable interfaces
5. Add/remove/modify services
6. etc...

The service types can be executed through the NetYCE API interfaces or from the NetYCE web-interface. Service types can create, modify or delete database objects, which can be located and used.

> Please consult the article Service-types overview for an introduction to Service-types
> A complete reference to all service type commands is listed in Service-type syntax.

Service-types form

The Service-types form is divided in 3 sections:

1. Service types
2. Rows within a selected service type
3. Details and editing of a selected row



Service-types section

Within the Service types section, the user has an overview of the service types which belong to the Client types to which the user has access. Here a new service type can be added , an existing service type can be deleted or duplicated .

Adding or duplicating a service type

When adding or duplicating a service type the following data must be entered:

- Client type (drop down menu)
- Service class (drop down menu). This list is build from the available service classes of the client type.
- The name of the service type. This is chosen by the user.
- The name of the service task. Create is a reserved value

Deleting a service type

When you want to delete a service type, simply select the service type to be deleted and then click the Delete button and the *OK* button.

Finding a service type

It is possible to search within the service types. Click the Search button and type the selection criteria in the search bar. Press **[enter]** to commit.

Rows

Within the second part of the window, the components of the selected service type are displayed.

Adding a row

When adding a new row, the new row will be added after the current selected row. To add a row click the button below the section. A new row will be added.

Deleting a row

In order to delete a row, click the icon below the section. You will be asked to confirm, click *OK* to delete the row.

Resequence rows

With the Up and Down buttons resequencing of the rows is possible. The aliases will not be updated!

Details of a row

When selecting a row, the details will be displayed in the third section of the form. After each updated value, the next field will be updated to ensure the accurate syntax. The description field can be used as comment. It is also possible to add an empty row and use this as a comment row. In this case only the description field will be set.

Alias

The **alias** is being used only internally within a Service type script to use the result of one row as a variable in another row. It is advisable to use distinctive names for aliases as well as surround them by <hooks> to identify them visually as such.

Please refer to Service-types overview for some guidelines on Alias names.

Syntax

The definition of a Service type is done with a specific syntax, which is guided by the visual editor. The **Add** and **Locate** can be assigned **aliases**, to be used as a reference to the object in the commands following it.

In general, everything written in UPPERCASE are syntax keywords, everything written in lowercase are **aliases** or **values**.

> The article on Service type syntax offers a summary of the service-type commands available. They are grouped around the various object types that NetYCE supports.

Example

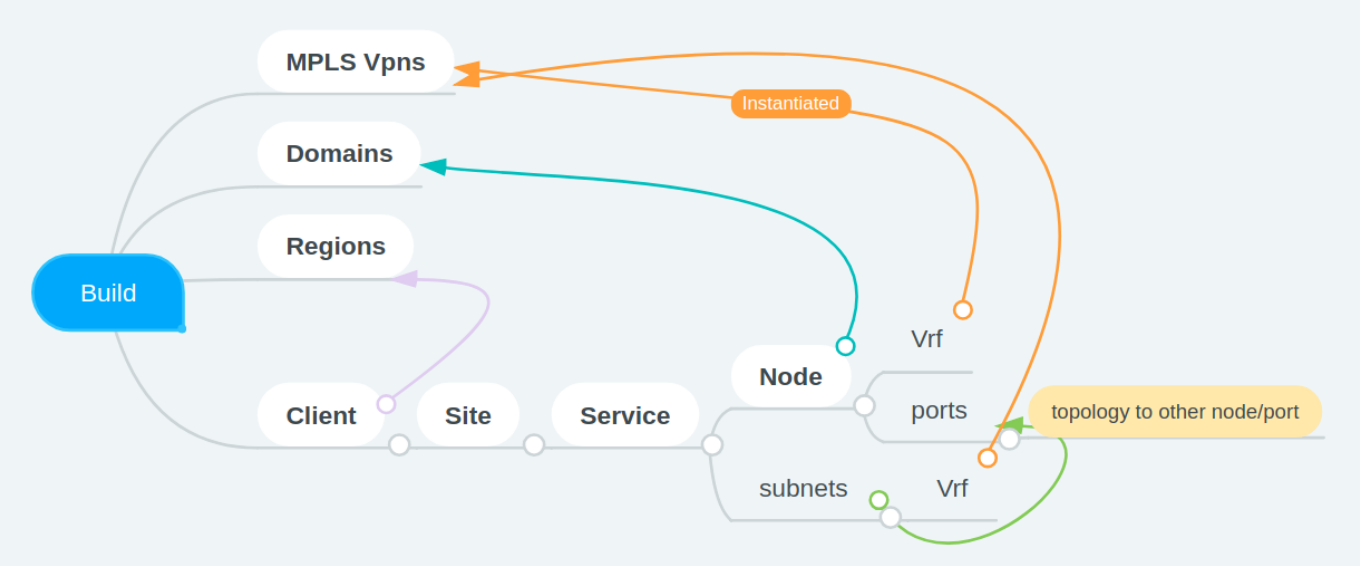
The following example will add a node to a newly created service with a IP address assigned to interface Loopback0.

Sequence	Exec	Class	Scope	Match	Value	Alias
1	ADD	SERVICE	SITE	CURRENT		<srv>
2	ADD	NODE	<srv>	NODE_TYPE	LAN_ACCESS	<node>
3	ADD	SUBNET	<srv>	NET_NAME	Loopback	
4	LOCATE	ADDRESS	ADDRESS_FIRSTFREE	Lo0	<ip>	
5	LOCATE	PORT	<node>	PORT_TEMPLATE_FIRST	Loopback0	<port>
6	ASSIGN	ADDRESS	<ip>	PORT	<port>	

A whole set of example Service types is included in every distribution. You can copy and adapt them to suit your own architecture.

Node and information structure

The below diagram shows how some of the database information for Nodes, Services, subnets, etc. is structured. This could help you design your service types and help you understand how each of the pieces tie together.



From:
https://wiki.netyce.com/ - **Technical documentation**

Permanent link:
https://wiki.netyce.com/doku.php?id=menu:design:service_types

Last update: **2024/07/03 12:31**